

No. Dole: 6766
Copy : 1

D3 658.503.6
Ron
S

**SISTEM PENDUKUNG KEPUTUSAN PROSES PERBAIKAN
DIES DENGAN METODE SMART MENGGUNAKAN
FRAMEWORK CODEIGNITER 3.1.8 DAN MySQL 5.5.32
PADA PT NUSA INDAH JAYA UTAMA**

TUGAS AKHIR

Diajukan Untuk Memenuhi Sebagian Syarat Penyelesaian Jenjang
Sarjana Terapan Program Studi Sistem Informasi Industri Otomotif
Pada Politeknik STMI Jakarta

OLEH

ABDUR RONI

1315062

DATA BUKU PERPUSTAKAAN	
Tgl Terima	20/07/22
No Induk Buku	473/5110/SB/TA/22



**POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA
JAKARTA**

2019

SUMBANGAN ALUMNI

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PENGESAHAN

JUDUL TUGAS AKHIR:

**SISTEM PENDUKUNG KEPUTUSAN PROSES PERBAIKAN *DIES*
DENGAN METODE *SMART* MENGGUNAKAN FRAMEWORK
CODEIGNITER 3.1.8 DAN MySQL 5.5.32 PADA
PT NUSA INDAH JAYA UTAMA**

Disusun Oleh:

Nama : Abdur Roni
NIM : 1315062
Program Studi : Sistem Informasi Industri Otomotif

Telah Diuji Oleh Tim Penguji Sidang Tugas Akhir Program Studi Sistem Informasi Industri Otomotif Politeknik STMI Jakarta Kementerian Perindustrian Republik Indonesia Pada Hari Kamis Tanggal 12 September 2019.

Jakarta, 17 September 2019

Dosen Pembimbing



Fifi L. Hadianastuti, S.Kom, M.Kes
NIP : 197310162005022001

Ketua Penguji



Lucky Heriyanto, S.T., M.T.I
NIP : 197908202009011009

Dosen Penguji



Dedy Trisanto, S.Kom, MMSI
NIP : 197805052005021002

Dosen Penguji



Ahlan Ismono, S.Kom, MMSI
NIP : 197901072006041002

POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN REPUBLIK INDONESIA

LEMBAR PERSETUJUAN DOSEN PEMBIMBING

JUDUL TUGAS AKHIR:

**“SISTEM PENDUKUNG KEPUTUSAN PROSES PERBAIKAN *DIES*
DENGAN METODE *SMART* MENGGUNAKAN FRAMEWORK
CODEIGNITER 3.1.8 DAN MySQL 5.5.32 PADA
PT NUSA INDAH JAYA UTAMA”**

DISUSUN OLEH:

Nama : Abdur Roni
NIM : 1315062
Program Studi : Sistem Informasi Industri Otomotif
Tanggal Seminar : 21 Agustus 2019
Tanggal Sidang : 12 September 2019
Tanggal Lulus : 12 September 2019

Jakarta, 17 September 2019

Dosen Pembimbing










Fifi Lailasari Hadianastuti, S.Kom, M.Kes

NIP : 197310162005022001

LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR

Nama : Abdur Roni
NIM : 1315062
Judul Tugas Akhir : Sistem Pendukung Keputusan Proses Perbaikan Dies
Dengan Metode SMART Menggunakan Framework
CodeIgniter 3.1.8 Dan MySQL 5.5.32 Pada
PT Nusa Indah Jaya Utama
Pembimbing : Fifi Lailasari Hadianastuti, S.Kom, M.Kes

Tanggal	Keterangan	Paraf
24 Juni 2019	Bimbingan Proposal Tugas Akhir	
25 Juni 2019	Revisi Bab I, II, III	
28 Juni 2019	Revisi Bab II, III dan Bimbingan Bab IV	
1 Juli 2019	Revisi Bab III dan IV	
8 Juli 2019	Revisi Bab IV	
17 Juli 2019	Bimbingan Bab V	
22 Juli 2019	Konsultasi Bab V	
29 Juli 2019	Revisi Bab V	
9 Agustus 2019	Revisi Bab V, Bimbingan Bab VI dan Demo Program	

Mengetahui,
Ketua Program Studi
Sistem Informasi Industri Otomotif



Noveriza Yuliasari, M.T
NIP : 197811212009012003

Dosen Pembimbing



Fifi Lailasari Hadianastuti, S.Kom, M.Kes
NIP : 197310162005022001

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini:

Nama : Abdur Roni

NIM : 1315062

Berstatus mahasiswa Program Studi Sistem Informasi Industri Otomotif di Politeknik STMI Jakarta Kementerian Perindustrian Republik Indonesia. Dengan ini menyatakan bahwa hasil karya Tugas Akhir yang saya buat dengan judul:

**“SISTEM PENDUKUNG KEPUTUSAN PROSES PERBAIKAN DIES
DENGAN METODE *SMART* MENGGUNAKAN FRAMEWORK
CODEIGNITER 3.1.8 DAN MySQL 5.5.32 PADA
PT NUSA INDAH JAYA UTAMA”**

- **Dibuat** dan selesaikan sendiri, dengan menggunakan literatur hasil kuliah, survey lapangan, dibantu oleh dosen pembimbing, serta buku-buku maupun jurnal-jurnal ilmiah yang menjadi bahan acuan yang tertera dalam referensi karya tugas akhir ini.
- **Bukan** merupakan hasil duplikasi karya tulis yang sudah dipublikasikan atau yang pernah dipakai sebelumnya untuk mendapatkan gelar sarjana di Universitas/ Perguruan Tinggi lain, kecuali yang telah disebutkan sumbernya dan dicantumkan pada referensi karya Tugas Akhir ini.
- **Bukan** merupakan karya tulis hasil terjemahan dari kumpulan buku atau jurnal acuan yang tertera dalam referensi karya Tugas Akhir ini.

Jika terbukti saya tidak memenuhi apa yang telah dinyatakan di atas, maka saya bersedia menerima sanksi atas apa yang telah saya lakukan sesuai dengan peraturan yang berlaku.

Jakarta, 17 September 2019



Abdur Roni

ABSTRAK

Divisi *maintenance* merupakan hal yang cukup dibutuhkan pada setiap perusahaan. Divisi *maintenance* mempunyai pengaruh terhadap proses produksi dimana apabila dalam suatu perusahaan tidak menjalankan *maintenance* dengan baik maka akan sangat mungkin terjadinya kerusakan pada mesin produksi yang membuat *line* produksi berhenti, dan apabila terdapat kerusakan mesin pada *line* produksi maka divisi *maintenance* harus langsung memperbaiki mesin yang rusak tersebut agar *line* produksi tidak terhenti lama. Dari penelitian yang dilakukan pada PT Nusa Indah Jaya Utama diperoleh beberapa permasalahan pada divisi *maintenance* khususnya *maintenance dies* dimana *staff* dari divisi *maintenance dies* kurang mengetahui proses apa saja yang akan dilakukan pada saat melakukan perbaikan *dies*. Untuk memecahkan permasalahan tersebut maka dibutuhkan sistem yang mampu mendukung keputusan dalam proses perbaikan *dies*. Sistem pendukung keputusan ini dibuat dengan menggunakan metode pengembangan *waterfall* dan metode sistem pendukung keputusan SMART, sistem ini dibangun juga dengan menggunakan *Framework CodeIgniter 3.1.8* dan *Data Base MySQL 5.5.32*. Dengan adanya sistem yang dibangun ini, maka diperoleh sistem pendukung keputusan proses perbaikan *dies* yang dapat membantu pengambilan keputusan dalam proses perbaikan *dies*.

Kata Kunci : Sistem Pendukung Keputusan, Perbaikan *Dies*, *Simple Multi Attribute Rating Technique (SMART)*, *Framework CodeIgniter 3.1.8*.

KATA PENGANTAR

Puji syukur kepada Allah SWT yang telah memberikan rahmat, karunia dan hidayahnya sehingga Tugas Akhir yang berjudul **“SISTEM PENDUKUNG KEPUTUSAN PROSES PERBAIKAN *DIES* DENGAN METODE *SMART* MENGGUNAKAN FRAMEWORK CODEIGNITER 3.1.8 DAN MySQL 5.5.32 PADA PT NUSA INDAH JAYA UTAMA”** dapat disusun dengan lancar. Tugas Akhir ini disusun sebagai syarat penyelesaian jenjang Sarjana Terapan program studi Sistem Informasi Industri Otomotif pada Politeknik STMI Jakarta.

Adapun tujuan dari Tugas Akhir ini ialah untuk membantu memecahkan permasalahan yang terjadi pada perusahaan khususnya pada bagian divisi *maintenance dies* dimana tempat penelitian berlangsung dengan memberikan usulan kepada perusahaan.

Dalam penyusunan Tugas Akhir ini, penulis banyak mendapat bantuan serta bimbingan dari berbagai pihak. Pada kesempatan ini tidak lupa penulis sampaikan ucapan terima kasih kepada:

1. Allah SWT atas rahmat serta kemudahan yang diberikan.
2. Kedua orang tua serta keluarga tercinta yang telah memberikan do'a, dukungan, pengorbanan, semangat dan kasih sayang hingga saat ini.
3. Bapak Dr. Mustofa, S.T., M.T., selaku Direktur Politeknik STMI Jakarta.
4. Ibu Noveriza Yuliasari, M.T selaku Ketua Program Studi Sistem Informasi Industri Otomotif.
5. Ibu Fifi Lailasari Hadianastuti, S.Kom, M.Kes selaku dosen wali dan juga dosen pembimbing yang telah memberikan arahan dan masukan dalam penyusunan Tugas Akhir.
6. Bapak H. Saipudin. Dipl. Eng selaku Direktur PT Nusa Indah Jaya Utama
7. Bapak Soetarman, Bapak Syarief Hidayat, Bapak Edi Supriyadi, Bapak Satiri, Bapak Sunarno, Bapak Acing Maulana dan seluruh pegawai PT Nusa Indah Jaya Utama yang telah membantu memberi arahan serta informasi dalam pengumpulan data yang dibutuhkan.

8. Seluruh dosen Politeknik STMI Jakarta yang telah memberikan ilmunya guna menambah pengetahuan dan wawasan yang sangat bermanfaat.
9. Teman-teman mahasiswa/i Politeknik STMI Jakarta Program Studi Sistem Informasi Industri Otomotif khususnya untuk teman-teman SA02 2015 atas kebersamaan, solidaritas, canda tawa serta motivasinya selama ini.
10. Teman-teman warung mbak yati dan warung mpok atas kebersamaan, solidaritas serta canda tawa yang diberikan selama masa perkuliahan.
11. Teman-teman praktik kerja lapangan yang biasa disebut dengan pejuang pekayon, atas semangat, kebersamaan, bantuan, solidaritas serta canda tawa.
12. Teman-teman pejuang tugas akhir yang biasa disebut dengan pejuang september, atas semangat, kebersamaan, bantuan, solidaritas serta canda tawa.
13. Bang Bana Maulana yang telah memberikan informasi tentang maintenance dies dan juga memberikan jamuannya setiap kali bertemu.
14. Frans Wiliam Jeremia Hasian, S.Tr.Kom yang telah membantu memberikan bimbingan serta petuahnya.
15. Semua pihak yang telah membantu dalam penulisan tugas akhir ini yang tidak bisa disebutkan satu persatu.

Semoga Allah SWT membalas kebaikan dan ketulusan semua pihak yang telah membantu menyelesaikan tugas akhir ini dengan melimpahkan rahmat dan karuniannya. Besar harapan Penulis bahwa tugas akhir ini dapat memberikan manfaat serta pengetahuan bagi pembacanya. Terima kasih.

Jakarta, 09 Agustus 2019

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	
LEMBAR PENGESAHAN	
LEMBAR PERSETUJUAN DOSEN PEMBIMBING	
LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR	
LEMBAR PERTANYAAN KEASLIAN	
ABSTRAK	
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR	xiii
DAFTAR TABEL.....	xv
DAFTAR LAMPIRAN.....	xvii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Pokok Permasalahan.....	2
1.3. Tujuan Penelitian.....	3
1.4. Batasan Masalah.....	3
1.5. Manfaat Tugas Akhir.....	4
1.6. Sistematika Penulisan.....	4
BAB II LANDASAN TEORI.....	6
2.1. Sistem	6
2.1.1. Bentuk Sistem.....	6
2.1.2. Jenis Sistem	6
2.1.3. Elemen Sistem	7
2.2. Keputusan.....	8
2.2.1. Tahapan Keputusan	8
2.3. Sistem Pendukung Keputusan (SPK).....	10
2.3.1. Komponen Sistem Pendukung Keputusan	11

2.3.2.	Karakteristik Sistem Pendukung Keputusan	12
2.4.	Perbaikan	13
2.5.	<i>DIES</i>	13
2.5.1.	Jenis-Jenis <i>Dies</i>	13
2.6.	<i>Fuzzy Multiple Attribute Decision Making</i>	14
2.6.1.	Metode <i>SMART</i>	14
2.6.1.1	Langkah – Langkah Metode <i>SMART</i>	15
2.6.1.2	Kelebihan Metode <i>SMART</i>	16
2.6.2.	Metode <i>SAW</i>	16
2.6.3.	Metode <i>AHP</i>	17
2.7.	<i>System Development Life Cycle (SDLC)</i>	17
2.7.1	Metode <i>Waterfall</i>	17
2.8.	Data	19
2.9.	<i>Flowmap</i>	20
2.10.	<i>Unified Model Language (UML)</i>	21
2.10.1.	<i>Use Case Diagram</i>	22
2.10.1.1	<i>Use Case Description</i>	23
2.10.2.	<i>Activity Diagram</i>	24
2.10.3.	<i>Sequence Diagram</i>	25
2.10.4.	<i>Class Diagram</i>	27
2.10.5.	<i>Deployment Diagram</i>	27
2.11.	Kamus Data	28
2.12.	Basis Data (<i>Database</i>)	29
2.13.	<i>Entity Relationship Diagram (ERD)</i>	30
2.14.	<i>Windows Navigation Diagram (WND)</i>	30
2.15.	<i>CodeIgniter</i>	31
2.16.	<i>MySQL (My Structured Query Language)</i>	32
2.17.	<i>XAMPP</i>	34
2.18.	<i>System Requirement</i>	34
BAB III	METODOLOGI PENELITIAN	36
3.1.	Metodologi Penelitian	36

3.2.	Jenis dan Sumber Data	36
3.3.	Metode Pengumpulan Data	37
3.4.	Metode Pengembangan Sistem	38
3.5.	Kerangka Penelitian	39
BAB IV	PENGUMPULAN DAN PENGOLAHAN DATA.....	43
4.1.	Sejarah Perusahaan.....	43
4.2.	Profil Perusahaan.....	44
4.3.	Visi dan Misi Perusahaan	45
4.3.1.	Visi.....	45
4.3.2.	Misi	45
4.4.	Struktur Organisasi.....	45
4.5.	<i>Job Description</i>	48
4.6.	Tenaga Kerja	52
4.7.	Lokasi Perusahaan	53
4.8.	Produk Perusahaan	54
4.9.	<i>Dies</i>	55
4.10.	Alat-Alat Yang Digunakan Dalam Proses Perbaikan.....	58
4.11.	<i>Job Description</i> Divisi <i>Maintenance Dies</i>	60
4.11.1.	Proses Perbaikan <i>Dies</i>	61
4.12.	Dokumen Terkait Proses Perbaikan <i>Dies</i>	63
4.13.	Pengolahan Data Kerusakan <i>Dies</i>	66
4.13.1	Kriteria dan Subkriteria	66
4.13.2	Contoh Kasus.....	67
BAB V	ANALISIS DAN PEMBAHASAN.....	72
5.1.	Analisis Kebutuhan Sistem	72
5.1.1	<i>Non Functional Requirement</i>	73
5.2.	Alur Sistem Pendukung Keputusan Proses Perbaikan	74
5.3.	Pemodelan Sistem Pendukung Keputusan	77
5.3.1.	<i>Use Case Diagram</i>	77
5.3.2.	<i>Use Case Description</i>	79
5.4.	<i>Activity Diagram</i>	84

5.5.	<i>Sequence Diagram</i>	91
5.6.	<i>Class Diagram</i>	98
5.7.	<i>Deployment Diagram</i>	99
5.8.	<i>Windows Navigation Diagram (WND)</i>	100
5.9.	Pemodelan Data Sistem Pendukung Keputusan Usulan...	102
5.9.1.	<i>Entity Relationship Diagram (ERD)</i>	102
5.9.2.	Kamus Data.....	103
5.10.	Perancangan Antarmuka Sistem Pendukung Keputusan...	
	Proses Perbaikan <i>Dies</i>	106
BAB VI	KESIMPULAN DAN SARAN	114
6.1.	Kesimpulan.....	114
6.2.	Saran	114
	Daftar Pustaka	115
	Lampiran	117

DAFTAR GAMBAR

	Halaman
Gambar II.1 Gambaran Sistem Terbuka	7
Gambar II.2 Gambaran Sistem Tertutup	7
Gambar II.3 Bagan Siklus Analisis Keputusan.....	9
Gambar II.4 Komponen Sistem Pendukung Keputusan	12
Gambar II.5 Metode <i>Waterfall</i>	18
Gambar II.6 Contoh Deployment Diagram.....	28
Gambar II.7 Contoh <i>Windows Navigation Diagram</i>	31
Gambar III.1 Kerangka Penelitian.....	42
Gambar IV.1 Logo PT Nusa Indah Jaya Utama.....	44
Gambar IV.2 Struktur Organisasi PT Nusa Indah Jaya Utama	47
Gambar IV.3 <i>Flowmap</i> Berjalan Proses Perbaikan <i>Dies</i>	62
Gambar IV.4 <i>Form Maintenance Dies Job Order</i>	63
Gambar IV.5 <i>Form List Dies</i>	64
Gambar IV.6 Buku Besar Pencatatan Hasil Perbaikan.....	65
Gambar V.1 <i>Flowmap</i> Sistem Pendukung Keputusan Proses Perbaikan.....	76
Gambar V.2 <i>Use Case Diagram</i> Sistem Pendukung Keputusan Usulan	78
Gambar V.3 <i>Activity Diagram</i> Mengelola Data <i>Master Dies</i>	85
Gambar V.4 <i>Activity Diagram</i> Mengelola Data <i>Master Rekanan</i>	86
Gambar V.5 <i>Activity Diagram</i> Mengelola Data <i>Master Kerusakan</i>	87
Gambar V.6 <i>Activity Diagram</i> Mengelola Hasil Perbaikan.....	88
Gambar V.7 <i>Activity Diagram</i> Menerima Permintaan Perbaikan.....	88
Gambar V.8 <i>Activity Diagram</i> Permintaan Perbaikan	89
Gambar V.9 <i>Activity Diagram</i> Konfirmasi Perbaikan Rekanan	89
Gambar V.10 <i>Activity Diagram</i> Permintaan Perbaikan Rekanan	90
Gambar V.11 <i>Activity Diagram</i> Melakukan <i>Login</i>	90
Gambar V.12 <i>Activity Diagram</i> Mencetak Data	91

Gambar V.13 <i>Sequence Diagram Login</i>	92
Gambar V.14 <i>Sequence Diagram</i> Mengelola Data <i>Dies</i>	92
Gambar V.15 <i>Sequence Diagram</i> Mengelola Data Kerusakan.....	93
Gambar V.16 <i>Sequence Diagram</i> Mengelola Data Rekanan	94
Gambar V.17 <i>Sequence Diagram</i> Menerima Perbaikan Rekanan	94
Gambar V.18 <i>Sequence Diagram</i> Menerima Permintaan Perbaikan	95
Gambar V.19 <i>Sequence Diagram</i> Permintaan Perbaikan Rekanan	96
Gambar V.20 <i>Sequence Diagram</i> Permintaan Perbaikan	96
Gambar V.21 <i>Sequence Diagram</i> Mengelola Hasil Perbaikan	97
Gambar V.22 <i>Sequence Diagram</i> Cetak Laporan	98
Gambar V.23 <i>Class Diagram</i> Sistem Usulan.....	99
Gambar V.24 <i>Deployment Diagram</i> Sistem Pendukung Keputusan Usulan..	100
Gambar V.25 <i>Windows Navigation Diagram</i> Sistem Usulan.....	101
Gambar V.26 <i>Entity Relationship Diagram</i> Sistem Usulan.....	102
Gambar V.27 <i>Form Login</i>	106
Gambar V.28 <i>Form</i> Halaman Utama	107
Gambar V.29 <i>Form</i> Data <i>Dies</i>	107
Gambar V.30 <i>Form</i> Tambah dan Ubah Data <i>Dies</i>	108
Gambar V.31 <i>Form</i> Data Kerusakan.....	108
Gambar V.32 <i>Form</i> Tambah dan Ubah Data Kerusakan	109
Gambar V.33 <i>Form</i> Data Rekanan.....	109
Gambar V.34 <i>Form</i> Tambah dan Ubah Data Rekanan	110
Gambar V.35 <i>Form Validasi</i> Perbaikan.....	110
Gambar V.36 <i>Form</i> Selesai Diperbaiki.....	111
Gambar V.37 <i>Form</i> Permintaan Perbaikan.....	111
Gambar V.38 <i>Form</i> Tambah Permintaan Perbaikan	112
Gambar V.39 <i>Form</i> Permintaan Disetujui	112
Gambar V.40 <i>Form</i> Laporan Perbaikan.....	113
Gambar V.41 <i>Form</i> Cetak Laporan Perbaikan	113

DAFTAR TABEL

	Halaman
Tabel II.1 Simbol-simbol Flowmap	20
Tabel II.2 Simbol-simbol <i>Use Case Diagram</i>	22
Tabel II.3 Elemen-elemen <i>Activity Diagram</i>	25
Tabel II.4 Elemen-elemen <i>Sequence Diagram</i>	26
Tabel II.5 Elemen-elemen <i>Class Diagram</i>	27
Tabel II.6 Contoh Kamus Data untuk Tabel Pemasok	28
Tabel II.7 Simbol-simbol ERD	30
Tabel II.8 Tipe Data Pada MySQL	33
Tabel IV.1 Produk Perusahaan PT Nusa Indah Jaya Utama	54
Tabel IV.2 <i>Dies</i> PT Nusa Indah Jaya Utama	56
Tabel IV.3 Alat-alat Yang Digunakan Dalam Proses Perbaikan	58
Tabel IV.4 Kriteria Penilaian Kerusakan <i>Dies</i>	66
Tabel IV.5 Subkriteria Penilaian Kerusakan <i>Dies</i>	67
Tabel IV.6 Alternatif	67
Tabel IV.7 Kriteria	68
Tabel IV.8 Nilai Parameter	68
Tabel IV.9 Kriteria Dari Setiap Alternatif	69
Tabel IV.10 Normalisasi Bobot	69
Tabel IV.11 Nilai Kriteria Dari Setiap Alternatif	69
Tabel IV.12 Hasil Perhitungan Dengan Metode SMART	70
Tabel IV.13 Hasil Keputusan	71
Tabel IV.1 Definisi Aktor Sistem Perbaikan <i>Dies</i>	69
Tabel IV.2 Definisi <i>Use Case</i> Sistem Perbaikan <i>Dies</i> Berjalan	70
Tabel V.1 Analisis Kebutuhan Sistem	72
Tabel V.2 <i>Use Case Description</i> Menerima Permintaan Perbaikan <i>Dies</i> ..	79

Tabel V.3	<i>Use Case Description</i> Membuat Permintaan Perbaikan Dies Pada Rekanan	79
Tabel V.4	<i>Use Case Description</i> Mengelola Hasil Perbaikan Dies	80
Tabel V.5	<i>Use Case Description</i> Mengelola List Dies	81
Tabel V.6	<i>Use Case Description</i> Mengelola Jenis Kerusakan Dies	81
Tabel V.7	<i>Use Case Description</i> Login	82
Tabel V.8	<i>Use Case Description</i> Mengelola Data Rekanan	83
Tabel V.9	<i>Use Case Description</i> Menerima Laporan Hasil Perbaikan Dies.....	83
Tabel V.10	<i>Use Case Description</i> Membuat Permintaan Perbaikan Dies ..	84
Tabel V.11	<i>Use Case Description</i> Membuat Permintaan Perbaikan Dies ..	84
Tabel V.12	Tabel User	103
Tabel V.13	Tabel <i>Dies Master</i>	103
Tabel V.14	Tabel <i>Dies Detail</i>	104
Tabel V.15	Tabel Rekanan	104
Tabel V.16	Tabel Kerusakan	104
Tabel V.17	Tabel Perbaikan	105
Tabel V.18	Tabel Perbaikan Rekanan	106

DAFTAR LAMPIRAN

	Halaman
Lampiran A Uraian Kegiatan Penelitian.....	L-1
Lampiran B Jadwal Observasi.....	L-3
Lampiran C Jadwal Wawancara	L-4
Lampiran D Transkrip Wawancara.....	L-5
Lampiran E <i>Listing Program</i>	L-7
Lampiran F Tampilan Program.....	L-15
Lampiran G <i>Black-Box Testing</i>	L-18

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam persaingan dunia industri saat ini, setiap perusahaan dituntut untuk dapat menghasilkan produk yang berkualitas serta memenuhi standar yang ada agar dapat bersaing terhadap perusahaan lainnya. Selain membuat produk yang berkualitas perusahaan juga harus dapat memberikan produk sesuai dengan kebutuhan pelanggan sehingga mampu memuaskan pelanggan lama serta dapat menarik pelanggan baru untuk menggunakan produk yang dihasilkan oleh perusahaan.

Agar perusahaan dapat menghasilkan produk yang berkualitas maka diperlukan kondisi mesin maupun alat-alat produksi yang baik, dikarenakan suatu produk dapat dihasilkan dengan baik apabila mesin ataupun alat-alat dalam proses produksi tersebut memiliki kondisi yang baik terlepas dari bahan baku dan sumber daya manusia yang mana juga harus memiliki kualitas yang baik.

Untuk menjaga mesin dan alat-alat produksi tetap dalam kondisi yang baik, maka diperlukan adanya divisi *maintenance* di dalam suatu perusahaan. Divisi *maintenance* pada perusahaan bertugas untuk melakukan proses perawatan dan perbaikan terhadap mesin serta alat-alat produksi agar kegiatan produksi yang dilakukan dapat berjalan dengan lancar dan juga dapat menghasilkan produk dengan standar dan kualitas yang baik.

PT Nusa Indah Jaya Utama merupakan perusahaan yang bergerak dibidang otomotif yang memproduksi *stamping parts* untuk kendaraan roda empat dan roda dua. Untuk menjaga kualitas produk yang dihasilkan dalam proses produksi, PT Nusa Indah Jaya Utama memiliki divisi *maintenance* dimana pada divisi ini dipecah menjadi dua bagian yaitu *maintenance machine* dan *maintenance dies*. Pemecahan bagian pada divisi ini dikarenakan perusahaan memproduksi *stamping parts* yang mana didalam proses produksi *stamping parts* terdapat dua bagian yang sangat penting yaitu mesin *press* dan juga *dies*.

Divisi *maintenance machine* bertugas untuk merawat serta memperbaiki mesin-mesin *press* dan mesin kompresor agar tetap dalam dalam kondisi yang baik sehingga kegiatan produksi dapat berjalan dengan lancar. Sama halnya dengan divisi *maintenance machine* divisi *maintenance dies* juga bertugas untuk merawat serta memperbaiki *dies* agar tetap dalam kondisi yang baik.

Pada PT Nusa Indah Jaya Utama terdapat beberapa permasalahan pada divisi *maintenance dies* yang mana seperti masih digunakannya buku besar sebagai media atau tempat untuk mencatat laporan hasil kegiatan perbaikan *dies* yang mana beresiko terjadinya kehilangan data, selain menggunakan buku besar pada PT Nusa Indah Jaya Utama terdapat juga adanya kendala dalam pengambilan keputusan dalam memperbaiki *dies* yang rusak dikarenakan divisi *maintenance dies* kurang mengetahui proses perbaikan apa saja yang akan dilakukan.

Untuk mengatasi masalah tersebut maka diperlukan adanya sebuah sistem yang terkomputerisasi agar dapat mempermudah kinerja dari divisi *maintenance dies*. Adapun judul dari Tugas Akhir ini ialah “SISTEM PENDUKUNG KEPUTUSAN PROSES PERBAIKAN *DIES* DENGAN METODE *SMART* MENGGUNAKAN *FRAMEWORK CODEIGNITER* 3.1.8 DAN *MySQL* 5.5.32 PADA PT NUSA INDAH JAYA UTAMA”

1.2 Pokok Permasalahan

Berdasarkan latar belakang yang telah diuraikan sebelumnya, maka dapat diketahui permasalahan-permasalahan yang ada pada proses perbaikan *dies* pada PT Nusa Inda Jaya Utama, diantaranya ialah sebagai berikut:

1. Terdapatnya kendala dalam pengambilan keputusan pada proses perbaikan *dies*, dikarenakan *staff* dari divisi *maintenance dies* kurang mengetahui proses perbaikan apa saja yang akan dilakukan apabila terjadi kerusakan.
2. Pencatatan hasil kegiatan perbaikan *dies* masih dilakukan pada buku besar sebagai media pencatatan, sehingga dapat mengakibatkan terjadinya kehilangan data dikarenakan rusaknya buku besar, baik itu

disebabkan oleh robeknya buku besar ataupun terkenanya oli dari tangan *staff* divisi *maintenance dies* pada saat melakukan pencatatan.

3. Buku besar tempat pencatatan hasil kegiatan perbaikan *dies* diberikan kepada *engineering manager* yang nantinya akan direkap sebagai laporan hasil perbaikan *dies*, dimana pada saat proses perekapan data hasil kegiatan perbaikan *dies* sering terjadinya keterhambatan dikarenakan tulisan yang terdapat pada buku besar sulit dibaca oleh *engineering manager*.

1.3 Tujuan Penelitian

Tujuan penelitian tugas akhir yang dilakukan terhadap divisi *maintenance dies* pada PT Nusa Indah Jaya Utama adalah untuk merancang dan membangun sistem informasi yang mampu:

1. Membangun sistem pendukung keputusan proses perbaikan *dies* yang mampu memperkuat divisi *maintenance dies* dalam pengambilan keputusan perbaikan *dies*.
2. Membangun sistem yang mampu mengelola data sesuai dengan apa yang dibutuhkan oleh divisi *maintenance dies*.
3. Membangun basis data untuk menyimpan hasil laporan kegiatan perbaikan *dies* yang dilakukan oleh divisi *maintenance dies*.

1.4 Batasan Masalah

Agar dalam penulisan tugas akhir ini dapat lebih fokus dan lebih terarah, maka diperlukan batasan-batasan sebagai berikut:

1. Tempat melakukan penelitian tugas akhir adalah pada PT Nusa Indah Jaya Utama.
2. Pengamatan dilakukan pada divisi *maintenance dies* selama satu bulan, mulai dari 6 Agustus sampai dengan 6 September 2018.
3. Ruang lingkup yang diamati sebatas pada divisi *maintenance dies* dalam proses perbaikan *dies*.

4. Sistem ini hanya bersifat rancang bangun, dan belum diimplementasikan dalam perusahaan.
5. Sistem ini dibangun dengan menggunakan *framework CodeIgniter* dan *MySQL*.

1.5 Manfaat Tugas Akhir

Adapun manfaat bagi perusahaan yang dapat diambil dari penelitian tugas akhir ini yaitu memiliki aplikasi pengambilan keputusan proses perbaikan *dies* yang sudah terintegrasi dengan basis data sebagai media penyimpanan dan pengelolaan data.

1.6 Sistematika Penulisan

Tugas Akhir ini disusun secara sistematis guna memberikan gambaran yang jelas mengenai isi dan pembahasan yang ada di dalamnya. Adapun sistematika penulisan tugas akhir ini terurai dalam enam bab, yaitu:

BAB I PENDAHULUAN

Pada bab ini memuat latar belakang, pokok permasalahan, tujuan penelitian, batasan masalah, manfaat tugas akhir dan sistematika penulisan.

BAB II LANDASAN TEORI

Pada bab ini membahas tentang berbagai teori yang diperoleh dari buku-buku, literatur, ataupun berbagai macam referensi yang berkaitan dengan tema yang diambil. Teori-teori yang dipaparkan pada laporan ini adalah seputar sistem pendukung keputusan, proses perbaikan *dies*, *Unified Modelling Language* (UML), dan teori-teori lain yang berhubungan dengan sistem pendukung keputusan proses perbaikan *dies*.

BAB III METODOLOGI PENELITIAN

Pada bab ini menjelaskan metode ilmiah dalam mencari, mengembangkan, dan menguji kebenaran tentang suatu pengetahuan. Selain itu dijelaskan pula kerangka pemecahan masalah yang menguraikan tahap-tahap untuk mengatasi permasalahan yang telah diidentifikasi.

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Pada bab ini akan membahas tentang data yang telah diperoleh berdasarkan penelitian di PT Nusa Indah Jaya Utama, meliputi profil perusahaan, struktur organisasi, dan dokumen yang terlibat pada proses perbaikan *dies*.

BAB V ANALISIS DAN PEMBAHASAN

Pada bab ini berisi analisis rinci dari pengolahan data, yakni mulai dari analisis kebutuhan sistem, pemodelan sistem dengan *Unified Modelling Language* (UML), pemodelan data dan *Entity Relationship Diagram* (ERD) dan perancangan antarmuka.

BAB VI KESIMPULAN DAN SARAN

Pada bab ini diuraikan kesimpulan berdasarkan hasil penelitian serta saran-saran dalam penerapan sistem untuk perusahaan dan pengembangan selanjutnya.

BAB II

LANDASAN TEORI

2.1 Sistem

Menurut Mcleod (dalam Yakub, 2012), Sistem adalah sekelompok elemen-elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan.

Suatu sistem pada dasarnya adalah sekelompok unsur yang erat hubungannya satu dengan yang lain yang berfungsi bersama-sama untuk mencapai tujuan tertentu. Sutabri (2012).

Sistem merupakan kumpulan dari beberapa bagian yang memiliki keterkaitan dan saling bekerja sama serta membentuk suatu kesatuan untuk mencapai tujuan dari sistem tersebut. Maksud dari sistem adalah untuk mencapai tujuan dan sasaran dalam ruang lingkup yang sempit (Rusdiana dan Irfan, 2014).

2.1.1 Bentuk Sistem

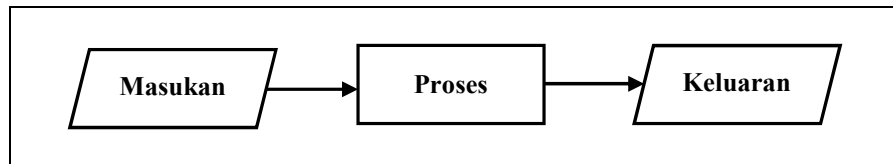
Rusdiana dan Irfan (2014), menjelaskan bahwa pada dasarnya hanya ada dua sistem, yaitu:

1. Sistem alami, seperti sistem matahari, sistem luar angkasa, sistem reproduksi dan sebagainya. Sistem alami dibagi menjadi dua, yaitu:
 - a. Sistem fisik, seperti sistem molekul, luar angkasa.
 - b. Sistem kehidupan, seperti sistem tumbuhan, sistem manusia.
2. Sistem buatan manusia, seperti sistem hukum, sistem perpustakaan, sistem transportasi dan sebagainya. Adapun sistem buatan manusia umumnya dibagi berdasarkan spesifikasi tertentu.

2.1.2 Jenis Sistem

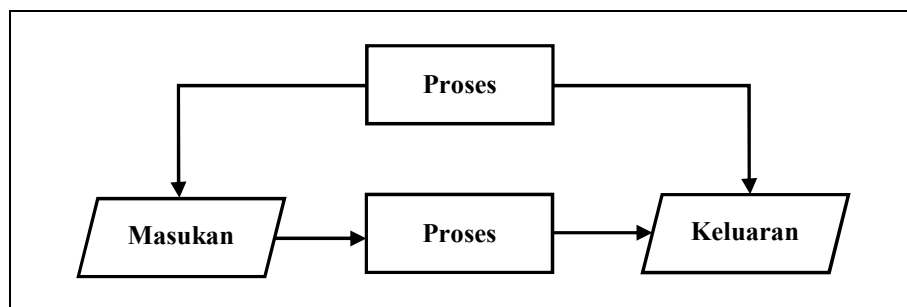
Jenis sistem secara umum terdiri atas sistem terbuka dan sistem tertutup. Sistem terbuka adalah sistem yang tidak memiliki sasaran, pengendalian mekanis, dan umpan balik. Adapun sistem yang tertutup adalah sebuah sistem yang

memiliki sasaran, pengendalian mekanis, dan umpan balik (Rusdiana dan Irfan, 2014).



Gambar II.1 Sistem Terbuka
(Sumber: Rusdiana dan Irfan, 2014)

Sistem tertutup adalah sistem yang tidak berinteraksi secara langsung dengan lingkungannya melalui arus sumber daya. Skema sistem tertutup dapat dilihat pada gambar di bawah ini.



Gambar II.2 Sistem Tertutup
(Sumber: Rusdiana dan Irfan 2014)

2.1.3 Elemen Sistem

Menurut Togar (1994) elemen sistem adalah bagian terkecil sistem yang dapat diidentifikasi. Jika sebuah sistem cukup besar yang terdiri dari sub-sistem, maka elemen sistem terdapat pada tingkat yang paling rendah yang dapat dikategorikan sebagai individu.

Elemen mempunyai ciri-ciri atau kualitas tersendiri, karena terdiri dari obyek atau manusia. Ciri-ciri ini mempunyai pengaruh terhadap operasi dari segi ketelitian, kecepatan, kapasitas dan lain-lain. Oleh karena itu pemilihan atribut harus relevan dengan tujuan dari sistem.

Menurut Kadir (2003) ada beberapa elemen yang membentuk sebuah sistem, yaitu:

1. Tujuan

Kegiatan yang menjadi pemotivasi yang mengarahkan sistem dengan adanya sistem tersebut dapat berjalan terarah dan terkendali.

2. Masukan (*input*)

Segala sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan untuk diproses.

3. Proses

Proses merupakan bagian yang melakukan perubahan atau transformasi dari masukan menjadi keluaran yang berguna.

4. Keluaran (*output*)

Keluaran merupakan hasil dari pemerosesan.

5. Mekanisme Pengendalian dan Umpan Balik

Mekanisme pengendalian (*control mechanism*) diwujudkan dengan menggunakan umpan balik (*feedback*) yang berupa keluaran. Umpan balik digunakan untuk mengendalikan baik masukan maupun proses dengan tujuan untuk mengatur agar sistem berjalan sesuai dengan tujuan

2.2 Keputusan

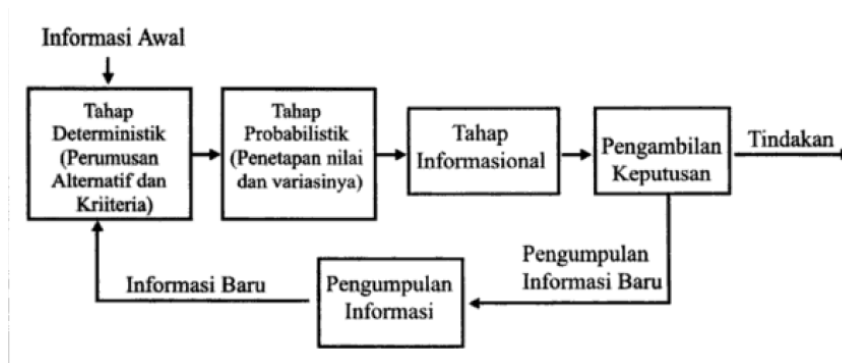
Keputusan sangat berkaitan erat dengan masalah. Masalah adalah suatu keadaan atau kondisi yang menjadi pemisah antara keadaan atau kondisi yang diinginkan dan keadaan atau kondisi kenyataan yang terjadi (Subri, 2011).

Menurut Kusri (2007 dalam Subri, 2011) keputusan adalah kegiatan memilih suatu strategi atau tindakan dalam pemecahan masalah. Tindakan memilih strategi atau aksi yang diyakini manajer akan memberikan solusi terbaik atas sesuatu itu disebut pengambilan keputusan.

2.2.1 Tahapan Keputusan

Mengambil atau membuat keputusan adalah suatu proses yang dilaksanakan orang berdasarkan pengetahuan dan informasi yang ada padanya pada saat tersebut dengan harapan bahwa sesuatu akan terjadi. Keputusan dapat

diambil dari alternatif-alternatif keputusan yang ada. Alternatif keputusan tersebut dapat dilakukan dengan adanya informasi yang diolah dan disajikan dengan sistem pendukung keputusan. Adapun informasi terbentuk dari adanya data yang terdiri dari bilangan dan *terms* yang disusun, diolah dan disajikan dengan dukungan sistem informasi manajemen. Kemudian keputusan yang diambil perlu ditindaklanjuti dengan aksi yang dalam pelaksanaannya perlu mengacu pada standar prosedur operasional (*Standard Operational Procedure*) dan akan membentuk kembali data, begitu seterusnya yang terjadi dalam siklus data, informasi, keputusan dan aksi (Faisal dan Permana, 2015).



Gambar II.3 Bagan Siklus Analisis Keputusan
(Sumber: Faisal dan Permana, 2015)

Menurut Marimin (2004 dalam Faisal dan Permana, 2015) dari informasi awal yang dikumpulkan, dilakukan pendefinisian dan penghubungan variabel-variabel yang mempengaruhi keputusan pada tahap *deterministic*. Setelah itu, dilakukan penetapan nilai untuk mengukur tingkat kepentingan variabel-variabel tersebut tanpa memperhatikan unsur ketidakpastian. Pada tahap *probabilistic*, dilakukan penetapan nilai ketidakpastian secara kuantitatif yang meliputi variabel-variabel yang sangat berpengaruh. Setelah didapatkan nilai-nilai variabel, selanjutnya dilakukan peninjauan terhadap nilai-nilai tersebut pada tahap informasional untuk menentukan nilai ekonomis pada variabel-variabel yang cukup berpengaruh, sehingga didapatkan suatu keputusan.

Keputusan yang dihasilkan dari tahap informasional dapat langsung ditindak lanjuti berupa tindakan, atau dapat dikaji ulang dengan mengumpulkan

informasi tambahan dengan tujuan untuk mengurangi kadar ketidak pastian. Dan jika hal ini terjadi, maka akan kembali lagi mengikuti ketiga tahap tersebut, begitu seterusnya (Faisal dan Permana, 2015).

2.3 Sistem Pendukung Keputusan (SPK)

Menurut Turban (2005 dalam Subri, 2011) secara formal sistem pendukung keputusan (SPK) tidak memiliki definisi yang tetap. Setiap definisi sistem pendukung keputusan (SPK) yang dibuat berusaha mempersempit ruang lingkungannya secara berbeda-beda. Sistem pendukung keputusan (SPK) adalah *computer based support system* untuk pengambilan keputusan manajemen yang berhubungan dengan kondisi keputusan semiterstruktur dan tidak terstruktur yang menggunakan berbagai model. Sistem pendukung keputusan (SPK) dapat digunakan oleh pengguna tunggal pada satu *personal computer* (PC) bisa juga berbasis *web* untuk digunakan oleh banyak orang yang terhubung melalui internet maupun intranet.

Sedangkan menurut Indriyani dan Humdiana (2005 dalam Subri, 2011) sistem pendukung keputusan (SPK) merupakan pengembangan lebih lanjut dari sistem informasi manajemen terkomputerisasi yang dirancang sedemikian rupa sehingga bersifat interaktif bagi pemakainya. Sifat interaktif ini dimaksudkan memudahkan integrasi antara berbagai komponen dalam proses pengambilan keputusan seperti prosedur, kebijakan, teknik analisis, serta pengalaman dan wawasan manajerial guna membentuk suatu kerangka keputusan. Pada umumnya, sistem pendukung keputusan (SPK) yang dirancang dapat memenuhi kemampuan, sebagai berikut:

1. Memberikan dukungan yang kuat bagi manajemen apabila pada suatu saat manajer dihadapkan dengan masalah-masalah yang sifatnya terstruktur maupun tidak terstruktur.
2. Memberikan dukungan pada proses pengambilan keputusan untuk semua tingkat manajemen dalam suatu perusahaan atau organisasi, dan mengintegrasikan semua tingkat manajemen pada saat yang tepat.

3. Memberikan dukungan komunikasi bagi para pengambil keputusan dalam rangka pengambilan suatu keputusan yang saling bergantung.
4. Mendukung semua langkah proses pengambilan keputusan dan memberikan fasilitas interaksi diantara langkah-langkah tersebut.
5. Mendukung berbagai proses pengambilan keputusan namun tidak menjadi seluruh proses manajerial tergantung pada SPK.
6. Mudah dalam pemakaiannya dan memungkinkan melakukan modifikasi terhadap perusahaan sesuai dengan perkembangan kebutuhan pengguna.

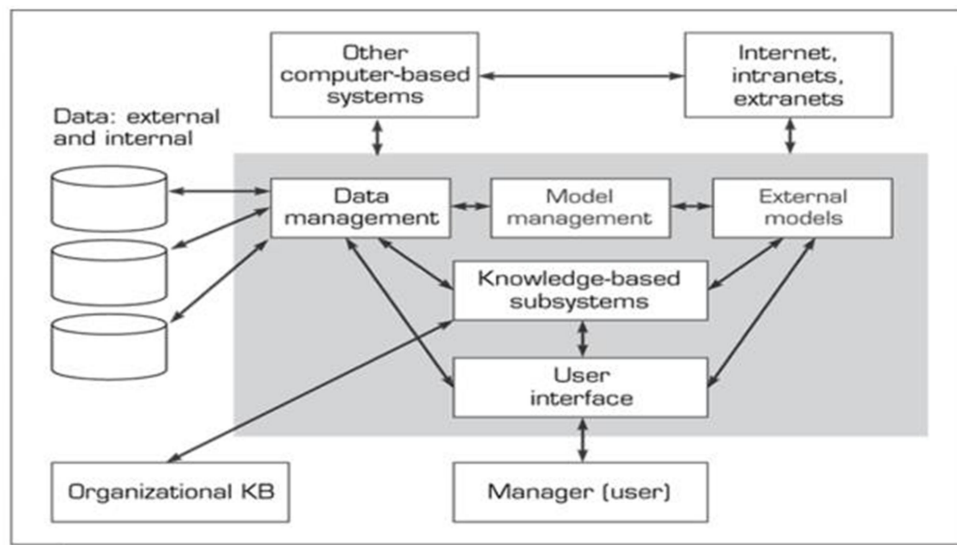
2.3.1 Komponen Sistem Pendukung Keputusan (SPK)

Menurut Turban (2005 dalam Subri 2011) bila dilihat dari karakteristik, sistem pendukung keputusan hanya mempunyai tiga komponen. Namun bagi tiga komponen tersebut memerlukan satu komponen lagi yang dapat memberikan intelegensi. Komponen keempat yaitu manajemen berbasis pengetahuan. Keempat komponen tersebut adalah:

1. **Subsistem Manajemen Data**
Komponen yang mengatur beragam aktivitas yang berhubungan dengan proses pengambilan, penyimpanan, dan pengorganisasian data yang relevan dengan konteks sebuah keputusan.
2. **Subsistem Manajemen Basis Model**
Komponen yang mengatur beragam aktivitas yang berhubungan dengan proses pengambilan, penyimpanan, dan pengorganisasian dari beragam model kuantitatif dalam keuangan, statistik, manajemen yang mempunyai kemampuan analisis terhadap masalah yang akan diambil keputusannya.
3. **Subsistem Antarmuka Pengguna (*User Interface*)**
Komponen sistem komputer yang memungkinkan komunikasi dua arah antara sistem komputer dan penggunanya.

4. Subsistem Manajemen Berbasis Pengetahuan

Melibatkan pengumpulan, pengelompokan dan penyebaran pengetahuan untuk mendukung kemampuan tertentu yang bermanfaat untuk memberikan intelegensi sesuai kebutuhan pemakai, biasanya berupa keahlian pakar. Komponen ini bersifat opsional.



Gambar II.4 Komponen Sistem Pendukung Keputusan (SPK)

(Sumber: Subri, 2011)

2.3.2 Karakteristik Sistem Pendukung Keputusan (SPK)

Sebagaimana dikatakan bahwa sistem pendukung keputusan berbeda dengan sistem informasi lainnya. Menurut Daihani (2001 dalam Subri 2011) beberapa karakteristik yang membedakannya adalah:

1. Sistem pendukung keputusan dirancang untuk membantu pengambilan keputusan dalam memecahkan masalah sifatnya semi terstruktur maupun tidak terstruktur.
2. Dalam proses pengolahannya, sistem pendukung keputusan mengombinasikan penggunaan model-model analisis dengan teknik pemasukan data konvensional serta fungsi-fungsi pencarian informasi.

3. Sistem pendukung keputusan, dirancang sedemikian rupa sehingga dapat digunakan dengan mudah oleh orang-orang yang tidak memiliki dasar kemampuan pengoperasian komputer tinggi.
4. Sistem pendukung keputusan dirancang dengan menekankan pada aspek fleksibilitas serta kemampuan adaptasi yang tinggi, sehingga mudah disesuaikan dengan berbagai perubahan lingkungan yang terjadi dan kebutuhan pemakai.

2.4 Perbaikan

Menurut Ansori dan Mustajib (2013) perbaikan merupakan pengujian secara menyeluruh dan perbaikan pada sedikit komponen atau seberapa besar komponen sampai pada kondisi yang dapat diterima. Perawatan perbaikan merupakan jenis perawatan yang terencana dan biasanya proses perawatannya dilakukan secara menyeluruh terhadap sistem, sehingga diharapkan sistem atau sebagian besar sub-sistem berada pada kondisi yang handal.

2.5 Dies

Dies atau *pressing dies* adalah suatu alat perkakas tekan yang digunakan untuk memotong atau membentuk suatu *sheet metal* sesuai dengan bentuk yang diinginkan. Dalam operasionalnya *pressing dies* ini digerakan oleh mesin *press* baik itu *mechanical press* maupun *hydraulics press machine*. (Azmi, 2011)

2.5.1 Jenis-Jenis Dies

Menurut azmi (2011) Secara umum konstruksi *dies* dibagi menjadi dua yaitu:

1. Single Operation Dies

Single operation dies atau biasa disebut *single dies* adalah konstruksi *dies* yang mempunyai sebuah proses pada *dies* setnya dan hanya menghasilkan sebuah *part* dalam sekali *stroke*. *Part* yang dihasilkan dapat berupa hasil *blank*, *part* setengah jadi atau *finish good part*.

2. *Multi Operation Dies*

Multi operation dies adalah *dies* yang didesain untuk bekerja pada dua atau lebih operasi dalam sekali *stroke*. Konstruksi *dies* ini memang lebih rumit sebab harus dicari kesesuaian *die height* dari proses-proses tersebut. Bila *die height*nya tidak sama maka *part* yang dihasilkan tidak sesuai dengan spesifikasi yang diinginkan dari *drawing* produknya.

2.6 *Fuzzy Multiple Attribute Decision Making*

Fuzzy Multiple Attribute Decision Making adalah suatu metode yang digunakan untuk mencari alternatif optimal dari sejumlah alternatif dengan kriteria tertentu. Inti dari FMADM adalah menentukan nilai bobot untuk setiap atribut, kemudian dilanjutkan dengan proses perankingan yang akan menyeleksi alternatif yang sudah diberikan. Pada dasarnya, ada 3 pendekatan untuk mencari nilai bobot atribut, yaitu pendekatan subyektif, pendekatan obyektif dan pendekatan Integrasi antara subyektif & obyektif.

Masing-masing pendekatan memiliki kelebihan dan kelemahan. Pada pendekatan subyektif, nilai bobot ditentukan berdasarkan subyektifitas dari para pengambil keputusan, sehingga beberapa faktor dalam proses perankingan alternatif bisa ditentukan secara bebas. Sedangkan pada pendekatan obyektif, nilai bobot dihitung secara matematis sehingga mengabaikan subyektifitas dari pengambil keputusan (Kusumadewi, 2007).

2.6.1 *Metode SMART*

SMART merupakan metode dalam pengambilan keputusan multi atribut. Teknik pengambilan multi atribut ini digunakan untuk mendukung pembuatan keputusan dalam memilih beberapa alternatif. Setiap pembuatan keputusan harus memiliki sebuah alternatif yang sesuai dengan tujuan yang dirumuskan. Setiap alternatif terdiri dari sekumpulan atribut dan setiap atribut mempunyai nilai dimana nilai ini dirata-ratakan dengan skala tertentu. Setiap atribut mempunyai bobot yang menggambarkan seberapa penting skala tertentu. Setiap atribut

mempunyai bobot yang menggambarkan seberapa penting suatu penting dibandingkan atribut lainnya. Pembobotan dan pemberian peringkat ini digunakan untuk menilai setiap alternatif agar diperoleh alternatif terbaik.

SMART menggunakan *linier adaptif model* untuk meramal nilai setiap alternatif. *SMART* lebih banyak digunakan karena kesederhanaannya dalam merespon kebutuhan pembuat keputusan dan caranya menganalisa respon. Analisis yang terbaik adalah transparan sehingga metode ini memberikan pemahaman masalah yang tinggi dan dapat diterima oleh pembuat keputusan. Pembobotan pada smart menggunakan skala 0 sampai 1, sehingga mempermudah perhitungan dan perbandingan nilai pada masing-masing alternatif (Yunitarini, 2013).

2.6.1.1 Langkah – Langkah Metode SMART

Berikut merupakan langkah-langkah penyelesaian metode SMART secara umum adalah sebagai berikut:

1. Menentukan masalah.
2. Menentukan kriteria yang akan digunakan.
3. Menentukan alternatif yang akan digunakan.
4. Memberikan bobot suatu kriteria yang akan digunakan.
5. Melakukan normalisasi dengan rumus:

$$\frac{w_j}{\sum w_j}$$

Keterangan : w_j = bobot suatu kriteria

6. Menghitung nilai utility untuk setiap kriteria masing-masing.

$$u_i(a_i) = \frac{(c_{out} - c_{min})}{(c_{max} - c_{min})}$$

Keterangan :

$u_i(a_i)$: nilai utility kriteria ke-i untuk kriteria ke-i

C_{max} : nilai kriteria maksimal

C_{min} : nilai kriteria minimal

C_{out} : nilai kriteria ke-i

7. Menghitung nilai akhir masing-masing.

2.6.1.2 Kelebihan Metode SMART

Metode SMART ini memiliki beberapa kelebihan dari metode lainnya diantaranya yaitu sebagai berikut:

1. Memungkinkan penambahan atau pengurangan alternatif.
2. Pada metode SMART penambahan atau pengurangan alternatif tidak akan mempengaruhi perhitungan pembobotan karena setiap penilaian alternatif tidak saling bergantung.
3. Perhitungan pada metode SMART lebih sederhana sehingga tidak diperlukan perhitungan matematis yang rumit dengan pemahaman matematika yang kuat.
4. Proses dalam menganalisa alternatif dan kriteria dalam SMART dapat dilihat oleh user, sehingga user dapat memahami bagaimana alternatif tertentu dapat dipilih. Alasan-alasan bagaimana alternatif itu dipilih dapat dilihat dari prosedur-prosedur yang dilakukan dalam SMART mulai dari penentuan kriteria, pembobotan dan pemberian nilai pada setiap alternatif.
5. Pembobotan yang dipakai dalam metode SMART ada 3 jenis, yaitu pembobotan secara langsung (*direct weighting*), pembobotan swing (*swing weighting*) dan pembobotan centroid (*centroid weighting*).

2.6.2 Metode SAW

Metode *Simple Additive Weighting* (SAW) sering juga di kenal dengan istilah metode penjumlahan berbobot. Konsep dasar metode *simple additive weighting* adalah mencari penjumlahan terbobot dari *rating* kinerja pada setiap alternatif pada semua atribut. Metode *simple additive weighting* disarankan untuk menyelesaikan masalah penyeleksian dalam sistem pengambilan keputusan multi proses. Metode *simple additive weighting* merupakan metode yang banyak digunakan dalam pengambilan keputusan yang memiliki banyak atribut. Metode *simple additive weighting* membutuhkan proses normalisasi matriks keputusan (x)

ke suatu skala yang dapat diperbandingkan dengan semua rating alternatif yang ada (Harold, 2015)

2.6.3 Metode AHP

Analytical hierarchy process (AHP) merupakan suatu model pendukung keputusan yang dikembangkan oleh Thomas L. Saaty. Metode AHP telah banyak digunakan untuk membantu pengambilan keputusan. *Analytic hierarchy process* (AHP) adalah teori pengukuran melalui perbandingan berpasangan dan tergantung pada penilaian dari para ahli untuk mendapatkan skala prioritas. Perbandingan yang dibuat menggunakan skala penilaian mutlak yang mewakili, berapa banyak lagi, satu elemen mendominasi yang lain sehubungan dengan atribut yang diberikan. Penilaian mungkin tidak konsisten, dan bagaimana mengukur inkonsistensi dan meningkatkan penilaian, bila mungkin untuk mendapatkan konsistensi yang lebih baik adalah kekhawatiran dari AHP. Skala prioritas yang berasal disintesis dengan mengalikan dengan prioritas *node* induk dan menambahkan untuk semua *node* tersebut.

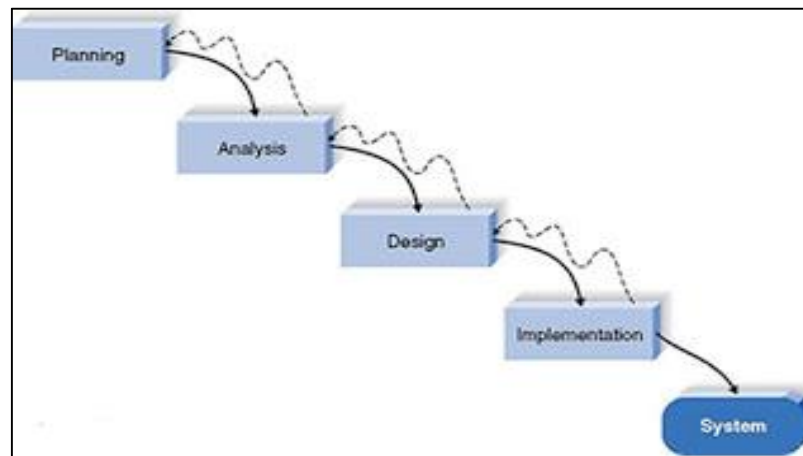
Model pendukung keputusan ini akan menguraikan masalah multi faktor atau multi kriteria yang kompleks menjadi suatu hirarki. Hirarki didefinisikan sebagai suatu representasi dari sebuah permasalahan yang kompleks dalam suatu struktur *multi-level* faktor, kriteria, sub kriteria, dan seterusnya ke bawah hingga *level* terakhir dari alternatif (Fahrozi, 2016).

2.7 *System Development Life Cycle* (SDLC)

Pengertian SDLC adalah suatu proses berkelanjutan dari *planning* analisis, desain dan implementasi. Yang pada prosesnya dilakukan perbaikan secara bertahap (Dennis *et al*, 2012).

2.7.1 Metode *Waterfall*

Model perancangan SDLC adalah dengan model *waterfall*. Terdapat 5 langkah dalam metode *waterfall*, yaitu sebagai berikut:



Gambar II.5 Metode *Waterfall*
(Sumber: Dennis, 2012)

a. Perancangan (*Planning*)

Pada tahap ini, menjelaskan dan mengargumentasikan untuk melanjutkan proyek yang telah dipilih. Rencana kerja yang matang juga disusun untuk menjalankan tahapan-tahapan lainnya. Pada tahap ini ditentukan secara detail rencana kerja yang harus dikerjakan, durasi yang diperlukan masing-masing tahap, sumber daya manusia, perangkat lunak, dokumentasi, perangkat keras, maupun financial diestimasi. Pembuatan perencanaan ini bukan langkah mudah karena untuk mengestimasi beban kerja dan durasi dari masing-masing tahap dibutuhkan pengalaman yang cukup banyak. Kesalahan pada tahap ini akan mengakibatkan keuntungan yang diperoleh tidak maksimal, bahkan bisa rugi.

b. Analisis

Tahap analisis adalah tahap dimana pengembang berusaha mengenali segala permasalahan yang muncul pada pengguna dengan mendekomposisi dan merealisasikan komponen-komponen sistem. Tujuan utama dari tahap analisis adalah untuk memahami dan mendokumentasikan kebutuhan bisnis dan persyaratan proses dari sistem baru. Menganalisis kebutuhan sebagai bahan dalam membuat spesifikasi di tahapan selanjutnya.

c. Desain

Tahap perancangan dimana pengembang mencari solusi permasalahan yang didapat dari tahap analisis. Tahapan mengubah kebutuhan yang masih berupa konsep menjadi spesifikasi sistem yang nyata untuk diimplementasikan.

d. Implementasi

Tahap implementasi dimana pengembang mengimplementasikan perancangan sistem ke situasi nyata atau desain harus diterjemahkan ke dalam bentuk mesin yang bisa dibaca. Disini mulai berurusan dengan pemilihan perangkat keras dan penyusunan perangkat lunak aplikasi (pengkodingan).

e. System

Pada tahapan sistem dilakukan pengujian (*testing*) dan pemeliharaan, yang dapat digunakan untuk menentukan apakah sistem atau perangkat lunak yang dibuat sudah sesuai dengan kebutuhan pengguna atau belum. Jika belum, proses selanjutnya adalah bersifat *interactive*, yaitu kembali ke tahap sebelumnya. Tahap pemeliharaan dan perawatan dimana kita mulai melakukan pengoperasian sistem dan jika diperlukan melakukan perbaikan-perbaikan kecil jika waktu pengguna sistem habis, maka akan lanjut kembali pada tahap perencanaan (*design*).

2.8 Data

Data adalah sesuatu yang belum mempunyai arti bagi penerimanya dan masih memerlukan adanya pengolahan. Data bisa berupa suatu keadaan, gambar, suara, huruf, angka, matematika, bahasa ataupun simbol-simbol lainnya yang dapat digunakan sebagai bahan untuk melihat lingkungan, objek, kejadian ataupun konsep (Rusdiana dan Irfan, 2014).

Sedangkan menurut Sutabri (2004) data adalah bahan mentah yang diproses untuk menyajikan informasi.

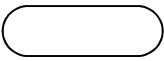

2.9 Flowmap

Menurut Jogiyanto (2005) *flowmap* atau bagan alir adalah bagan yang menunjukkan aliran di dalam program atau prosedur sistem secara logika. *Flowmap* ini berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *flowmap* ini harus dapat memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi. Adapun pedoman-pedoman dalam pembuatan *flowmap* adalah sebagai berikut:

1. *Flowmap* sebaiknya digambarkan dari atas ke bawah dan mulai dari bagian kiri dari suatu halaman.
2. Kegiatan di dalam *flowmap* harus ditunjukkan dengan jelas.
3. Harus ditunjukkan dari mana kegiatan akan dimulai dan dimana akan berakhir.
4. Masing-masing kegiatan di dalam *flowmap* sebaiknya digunakan suatu kata yang mewakili suatu pekerjaan.
5. Masing-masing kegiatan di dalam *flowmap* harus di dalam urutan yang semestinya.
6. Kegiatan yang terpotong dan akan disambung ditempat lain harus ditunjukkan dengan jelas menggunakan simbol penghubung.
7. Gunakan simbol simbol *flowmap* yang standar.

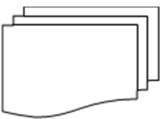
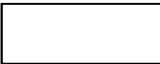
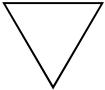


Adapun simbol-simbol yang sering digunakan dalam *flowmap* dapat dilihat pada tabel II.1 berikut ini:

Tabel II.1 Simbol-Simbol *Flowmap*

Simbol	Nama	Fungsi
	Mulai/Akhir	Menunjukkan dimulai/akhir dari sebuah proses.
	Dokumen	Simbol yang menyatakan <i>input</i> berasal dari dokumen dalam bentuk kertas atau <i>ouput</i> dicetak dikertas.

(Sumber: Jogiyanto, 2005)

Tabel II.1 Simbol-Simbol *Flowmap* (lanjutan)

Simbol	Nama	Fungsi
	Dokumen Rangkap	Menunjukkan <i>multi</i> dokumen.
	Proses	Proses pengolahan data.
	Arsip	Menunjukkan pengarsipan data.
	Garis Alir	Arus dari suatu proses.
	<i>Input/Output</i>	Simbol <i>input/output</i> digunakan untuk mewakili data <i>input/output</i>

(Sumber: Jogiyanto, 2005)

2.10 *Unified Modeling Language (UML)*

Unified modeling language (UML) adalah bahasa yang telah menjadi standar dalam industri untuk *visualisasi*, merancang dan mendokumentasikan sistem perangkat lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem.

Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram peranti lunak. Setiap bentuk memiliki warna tertentu, dan UML *syntax* mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Notasi UML terutama diturunkan dari 3 notasi yang telah ada sebelumnya, yaitu Grady Booch OOD (*Object-Oriented Design*), James Rumbaugh OMT (*Object Modeling Technique*) dan Ivar Jacobson OOSE (*Object-Oriented Software Engineering*).

UML didefinisikan sebagai bahasa visual untuk menjelaskan, memberikan spesifikasi, merancang, membuat model dan mendokumentasikan aspek-aspek dari sebuah sistem. Karena tergolong bahasa visual, UML lebih mengedepankan




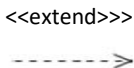
penggunaan diagram untuk menggambarkan aspek dari sistem yang sedang dimodelkan.

Pemodelan menggunakan UML merupakan metode pemodelan berorientasi objek dan berbasis visual. Oleh karena itu, pemodelan menggunakan UML merupakan pemodelan objek yang fokus pada pendefinisian struktur statis dan model sistem informasi yang dinamis daripada mendefinisikan data dan model proses yang tujuannya adalah pengembangan tradisional.

2.10.1 Usecase Diagram

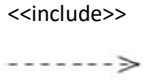
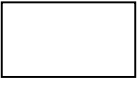

Diagram *use case* dibuat untuk menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Suatu landasan pola pikir yang ditekankan dalam diagram *use case* ini adalah apa yang dapat diperbuat oleh sistem, dan bukan bagaimana sistem melakukannya. Diagram *use case* menggambarkan kebutuhan sistem dari sudut pandang *user*, yaitu proses yang dilakukan oleh sistem dalam melayani *user* yang berinteraksi dengan sistem tersebut. Simbol-simbol yang terdapat didalam *use case diagram* dijelaskan pada tabel II.2 dibawah ini:

Tabel II.2 Simbol-simbol *Use case Diagram*

No	Simbol	Nama	Deskripsi
1		<i>Use case</i>	Merepresentasikan bagian utama dari sistem secara fungsional.
2		<i>Actor</i>	Seseorang atau sistem yang mendapatkkn keuntungan dari sistem
3		<i>Association</i>	Menghubungkan suatu <i>actor</i> dengan <i>use case</i> .
4		<i>Extend</i>	Merepresentasikan eksistensi dari <i>use case</i> untuk menyertakan perilaku <i>optional</i> .

(Sumber: Dennis, 2012)

Tabel II.2 Simbol-simbol *Use case Diagram* (lanjutan)

No	Simbol	Nama	Deskripsi
5		<i>Include</i>	Mewakili dimasukkannya satu <i>use case</i> ke <i>use case</i> yang lain.
6		<i>Subject</i>	Menyertakan nama subjek didalam maupun diatas
7		<i>Generalization</i>	Merepresentasikan <i>use case</i> khusus ke yang lebih umum.

(Sumber: Dennis, 2012)

2.10.1.1 Use Case Description

Use case description merupakan deskripsi yang mencatat mengenai detail pemrosesan dari suatu *use case* (Satzinger et al, 2010). *Use case description* memiliki urutan yang lengkap dari tahapan-tahapan untuk menyelesaikan suatu proses bisnis. Menurut Dennis et al (2012), elemen yang dimiliki *use case description* adalah sebagai berikut:

1. Basic Information

a. Nama

Nama digunakan untuk menjelaskan *use case* dan membedakan *use case* satu dengan yang lainnya. Nama yang digunakan harus sederhana namun mampu menjelaskan kondisi yang memungkinkan.

b. Description

Description berarti uraian singkat untuk menyampaikan tujuan dari *use case*.

c. Aktor

Aktor merujuk kepada orang, sistem, perangkat lunak ataupun perangkat keras yang berinteraksi langsung dengan sistem untuk mencapai tujuan.

2. *Normal Courses*

Bagian utama dari *use case* berikutnya adalah *normal courses*. Pada *normal courses* menjelaskan mengenai langkah-langkah yang dilakukan untuk mengeksekusi respon terhadap masukan dan keluaran.

3. *Alternative Courses*

Untuk beberapa *use case* memiliki jalur alternatif pada setiap kasusnya. Pada penggambaran jalur alternatif ini terdapat cabang yang juga akan menghasilkan kesimpulan dari *use case* tersebut.

2.10.2 *Activity Diagram*



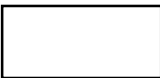




Activity diagram digunakan untuk memodelkan perilaku dalam proses bisnis yang independen terhadap objek. Dalam banyak hal, *activity diagram* dapat dipandang sebagai data *flow diagram* canggih yang digunakan pada konjungsi dengan analisis terstruktur. Walaupun begitu, *activity diagram* tidak seperti data *flow diagram*, *activity diagram* menggunakan notasi yang mengalamatkan pemodelan paralel. (Dennis, 2012).

Yang perlu diperhatikan disini adalah bahwa diagram aktifitas menggambarkan aktifitas sistem bukan apa yang dilakukan aktor, jadi aktifitas yang dapat dilakukan oleh sistem. Diagram aktifitas juga banyak digunakan untuk mendefinisikan hal-hal berikut:

- a. Rancangan proses bisnis dimana setiap urutan aktifitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan
- b. Urutan atau pengelompokan tampilan dari sistem/*user interface* dimana setiap aktifitas dianggap memiliki sebuah rancangan antarmuka tampilan
- c. Rancangan pengujian dimana setiap aktifitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya
- d. Rancangan menu yang akan ditampilkan pada perangkat lunak

Berikut adalah simbol-simbol yang ada pada *activity diagram*:

Tabel II.3 Elemen-elemen *Activity Diagram*

No	Simbol	Nama	Deskripsi
1.		<i>Action</i>	Merupakan notasi yang simpel dan tidak untuk dianalisis perilakunya. Dinamakan sesuai aksi yang dilakukan.
2.		<i>Activity</i>	Digunakan untuk mewakili satu set aksi yang dilakukan.
3.		<i>Object Node</i>	Digunakan untuk mewakili objek yang terhubung ke satu set arus objek.
4.		<i>Control Flow</i>	Menampilkan urutan eksekusi.
5.		<i>Object Flow</i>	Menunjukkan aliran dari sebuah objek dari satu aktifitas ke aktifitas lainnya.
6.		<i>Initial Node</i>	Menggambarkan permulaan dari suatu set aktifitas atau aksi.
7.		<i>Final-activity Node</i>	Menggambarkan akhir dari setiap aktifitas.





(Sumber: Dennis et al, 2012)

2.10.3 *Sequence Diagram*

Sequence diagram adalah salah satu dari dua jenis diagram interaksi. *Sequence diagram* menggambarkan benda-benda yang berpartisipasi dalam kasus penggunaan dan pesan yang melewati antara mereka dari waktu ke waktu untuk satu *use case*. *Sequence diagram* adalah model dinamis yang menunjukkan urutan eksplisit pesan yang lewat di antara objek dalam interaksi didefinisikan.

Karena urutan diagram menekankan pemesanan berbasis waktu kegiatan yang terjadi di antara set benda, diagram tersebut sangat membantu untuk memahami spesifikasi *real-time* dan kompleks menggunakan kasus (Dennis, 2012). Berikut simbol yang ada pada *Sequence Diagram*:

Tabel II.4 Elemen-elemen *Sequence Diagram*

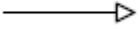
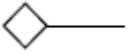

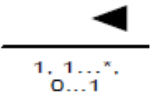
No	Simbol	Nama	Deskripsi
1.	 Actor	Aktor	Orang atau sistem yang berasal dari manfaat dan eksternal ke sistem yang berpartisipasi secara berurutan dengan mengirim dan atau menerima pesan.
2.		<i>Lifeline</i>	Menyatakan kehidupan suatu objek.
3.		Objek	Berpartisipasi secara berurutan dengan mengirim dan atau menerima pesan yang ditempatkan diatas diagram.
4.		<i>Execution Occurence</i>	Menyatakan objek dalam keadaan aktif dan berinteraksi pesan.
5.		<i>Message</i>	Pesan yang menggambarkan komunikasi yang terjadi antar objek.
6.		<i>Message (return)</i>	Pesan yang dikirim untuk diri sendiri secara langsung.
7.		<i>Message (return)</i>	Pesan yang dikirim untuk diri sendiri.
8.	X	<i>Object Destruction</i>	Sebuah x ditempatkan di ujung suatu objek garis hidup untuk menunjukkan bahwa akan keluar dari eksistensi.
9.	<i>Context</i>	<i>Frame</i>	Menunjukkan konteks dari <i>sequence diagram</i> .

(Sumber: Dennis el al, 2012)

2.10.4 Class Diagram

Sebuah diagram kelas adalah model statis yang menunjukkan kelas dan hubungan antar kelas yang tetap konstan dalam sistem dari waktu ke waktu. Sebuah *class* merupakan deskripsi dari sekumpulan objek yang memiliki *property* (*attribute*), operasi (*method*), relasi (*association*), dan tingkah laku (*behavior*) yang sama. Sebuah class menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi) (Alan Dennis, 2012). Berikut simbol-simbol yang ada pada diagram kelas:

Tabel II.5 Elemen-elemen *Class Diagram*

No	Simbol	Nama	Deskripsi
1.		<i>Generalization</i>	Merupakan sebuah <i>taxonomic relationship</i> antara <i>class</i> yang lebih umum dengan <i>class</i> yang lebih khusus.
2.		<i>Aggregation</i>	Menggambarkan suatu <i>class</i> terdiri dari <i>class</i> lain atau suatu <i>class</i> adalah bagian dari <i>class</i> lain.
3.		<i>Class</i>	Kelas pada struktur sistem.
4.		<i>Association</i>	Asosiasi yang menghubungkan <i>class</i> dengan <i>class multiplicity</i> .

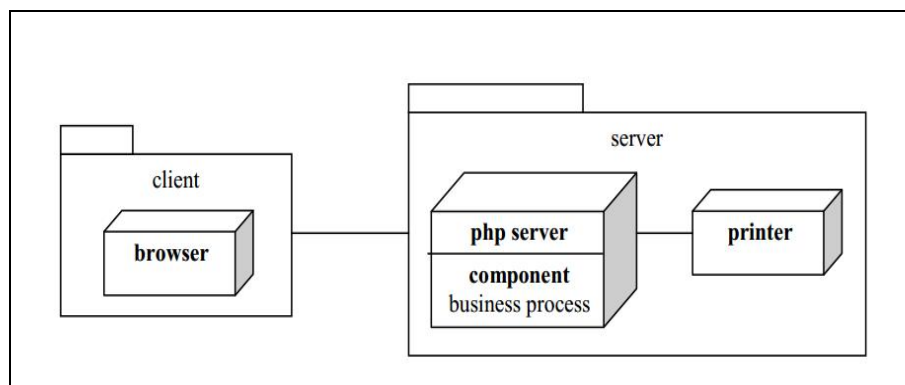
(Sumber: Dennis et al, 2012)

2.10.5 Deployment Diagram

Menurut A.S Rosa dan M. Shalahuddin (2013), diagram *deployment* atau *deployment diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. Diagram *deployment* juga dapat digunakan untuk memodelkan hal-hal berikut.

- Sistem tambahan (*embedded system*) yang menggambarkan rancangan *device*, *node* dan *hardware*.
- Sistem *client* atau *server*.
- Sistem terdistribusi murni.

Contoh dari *deployment* diagram dapat dilihat pada Gambar II.6.



Gambar II.6 Contoh *Deployment* Diagram
(Sumber: A.S Rosa, 2013)

2.11 Kamus Data

Menurut Jogiyanto (2005) kamus data (*data dictionary*) adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Dengan menggunakan kamus data diharapkan, analisis sistem dapat mendefinisikan data yang mengalir dalam sistem dengan lengkap. Kamus data dibuat pada tahap analisis sistem dan digunakan baik pada tahap analisis maupun pada tahap perancangan sistem.

Kamus data dapat berfungsi membantu pelaku sistem untuk mengartikan aplikasi secara detail dan mengorganisasi semua elemen data yang digunakan di dalam sistem secara persis sehingga pemakai dan penganalisis sistem mempunyai dasar pengertian yang sama tentang masukan, keluaran, penyimpanan dan proses. Berikut adalah contoh penulisan kamus data:

Spesifikasi tabel pemasok

Nama tabel : Pemasok

Tipe : File *master*

Tabel II.6 Contoh Kamus Data Untuk Tabel Pemasok

No	Nama Elemen	Akronim	Tipe	Width	Keterangan
1.	ID pemasok	ID_pemasok	<i>Varchar</i>	10	<i>Primary Key</i>
2.	Nama pemasok	Nama_pemasok	<i>Char</i>	40	

(Sumber: Jogiyanto, 2005)

Tabel II.6 Contoh Kamus Data Untuk Tabel Pemasok (lanjutan)

No	Nama Elemen	Akronim	Tipe	Width	Keterangan
3.	Alamat pemasok	Alamat	<i>Varchar</i>	100	
4.	Nomor telepon	Telepon	<i>Varchar</i>	12	

(Sumber: Jogiyanto, 2005)

2.12 Basis Data (*Database*)

Basis data terdiri dari dua kata, yaitu basis dan data. Basis kurang lebih dapat diartikan sebagai marka atau gudang, tempat bersarang/berkumpul, sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep dan sebagainya yang diwujudkan dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya.



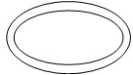
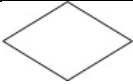
Menurut Bambang Hariyanto (2004) basis data adalah kumpulan data (elementer) yang secara logik berkaitan dalam merepresentasikan fenomena/fakta secara terstruktur dalam domain tertentu untuk mendukung aplikasi pada sistem tertentu. Basis data adalah kumpulan data yang saling berhubungan yang merefleksikan fakta-fakta yang terdapat di organisasi. Menurut Fatansyah (2015) sebagai satu kesatuan istilah, basis data (*database*) sendiri dapat didefinisikan dalam sejumlah sudut pandang seperti:

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (*redundancy*) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
3. Kumpulan *file*/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

2.13 *Entity Relationship Diagram (ERD)*

Berikut Terdapat sebuah model perancangan hubungan antar entitas (tabel) dari sebuah basis data (*database*). Istilah untuk frase ini biasa dikenal dengan nama *Entity Relationship Model*. Model hubungan ini seterusnya akan berlanjut menjadi sebuah diagram hubungan antar entitas yang biasa dikenal dengan nama *entity relationship diagram (ERD)* (Jeffrey A. Hoffer, 2007). Simbol-simbol ERD dapat dilihat pada Tabel II.7.

Tabel II.7 Simbol-Simbol ERD

Simbol	Nama	Deskripsi
	<i>Strong Entity</i>	Setiap hal dunia nyata (orang, tempat, objek, konsep, aktivitas) tentang suatu perusahaan mencatat data.
	<i>Attribute</i>	Properti atau karakteristik tipe entitas.
	<i>Multivalued Attribute</i>	Karakteristik tipe entitas yang dapat memiliki nilai lebih dari satu.
	<i>Relationship</i>	Hubungan antar entitas.

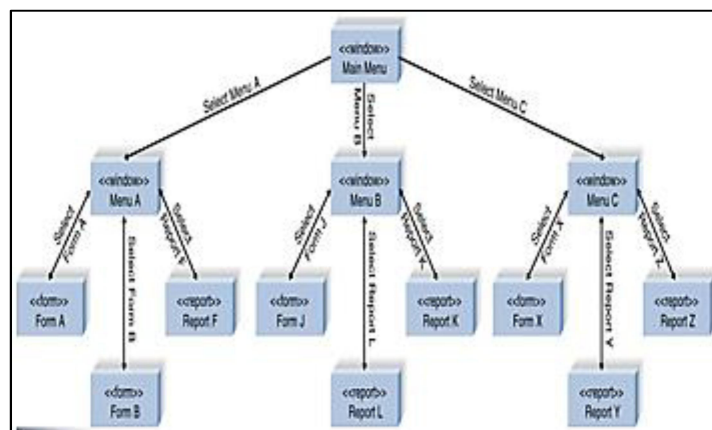
(Sumber: Jeffrey A. Hoffer, 2007)

2.14 *Windows Navigation Diagram (WND)*

Windows navigation diagram merupakan *statechart* diagram khusus yang berfokus pada *user interface*. *Windows navigation diagram (WND)* menurut Dennis (2012) adalah sebagai berikut:

1. Menunjukkan bagaimana semua layar, formulir, dan laporan terkait.
2. Menunjukkan bagaimana pengguna bergerak dari satu ke yang lain.
3. Seperti diagram keadaan untuk *user interface*.
 - a. Kotak mewakili komponen.
 - b. Panah mewakili transisi.
 - c. Stereotipe menunjukkan tipe antarmuka.

Contoh dari *windows navigation diagram* dapat dilihat pada Gambar II.7.



Gambar II.7 Contoh *Windows Navigation Diagram*
(Sumber: Dennis, 2012)

2.15 *CodeIgniter*

CodeIgniter adalah *framework* aplikasi web yang bersifat *open source* yang digunakan untuk membangun aplikasi PHP yang dinamis. Tujuan utamanya adalah untuk membantu pengembangan pengembang *CodeIgniter* untuk mengerjakan aplikasi lebih cepat daripada menulis semua kode dari nol.

CodeIgniter memiliki karakteristik yang fleksibel dan ringan untuk memudahkan dalam belajar, memodifikasi, dan mengintegrasikan *Library and Helper*. Menggunakan pola MVC sehingga struktur kode yang dihasilkan lebih terstruktur dan memiliki standar yang jelas.

Struktur kerja *CodeIgniter* dimulai dari browser yang akan berinteraksi via *controller*. Kemudian *controller* akan menerima dan membalas semua permintaan dari *browser*. Untuk data, *controller* akan meminta model dan untuk pengontrol UI/*template* akan meminta untuk melihat. Bila *browser* meminta halaman web maka *router* akan menemukan *controller* yang seharusnya menangani permintaan tersebut. Nantinya akan digunakan *controller* untuk mengakses data dan tampilan model untuk menampilkan data.

MVC digunakan untuk memisahkan akses data dan logika bisnis dari presentasi data dan interaksi pengguna. Pemisahan dilakukan agar setiap perubahan dalam logika presentasi atau logika bisnis tidak berpengaruh satu sama lain sangat kompleks. Solusi pemisahan MVC diharapkan dapat meningkatkan

fleksibilitas dan usabilitas aplikasi. Arsitektur MVC memisahkan aplikasi menjadi tiga bagian, yaitu *model*, *view* dan *controller*.

1. *Model*

Representasi *database*, termasuk tabel desain terhadap hubungan yang ada antar tabel. Fungsi utama dari model ini adalah untuk menangani data, mengambil data dari *database*, memasukkan data ke dalam *database*, manipulasi data melalui validasi data.

2. *View*

View untuk membuat model data yang diberikan dan akan mengirimkan gerak/aktivitas dari konsistensi tampilan data ke perubahan yang terjadi. Dengan mengelompokkan semua tampilan dan kode presentasi di satu tempat, akan memudahkan untuk mengubah tampilan tanpa mempengaruhi logika bisnis dan data.

3. *Controller*

Controller mendefinisikan perilaku yang terjadi pada aplikasi, dan kemudian memetakannya ke dalam tindakan dari pengguna ke model. Pengontrol akan sangat erat kaitannya dengan *view*, karena setiap interaksi pengguna yang akan dilakukan akan ditunjukkan oleh *view* to a response oleh *Controller*. Di dalam *controller* akan ada metode yang akan merespon perilaku aplikasi.

CodeIgniter adalah sebuah *framework* PHP yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi *web* berbasis PHP dibandingkan jika menulis semua kode program dari awal. CodeIgniter menyediakan banyak *library* untuk mengerjakan tugas-tugas yang umumnya ada pada sebuah aplikasi berbasis *web*. Selain itu, struktur dan susunan logis dari *CodeIgniter* membuat aplikasi yang dibuat menjadi semakin teratur dan rapi.

2.16 MySQL (*My Structured Query Language*)

MySQL adalah sebuah program pembuat dan pengelola *database* atau yang sering disebut dengan DBMS (*Database Management System*), sifat dari DBMS ini adalah *open source* (Nugroho, 2007).

MySQL adalah suatu bahasa permintaan yang terstruktur dan telah distandarkan untuk semua program pengakses *database* seperti Oracle, PostgreSQL, SQL Server, dan lain-lain (Kadir, 2008).

Pada MySQL masing-masing tipe data memiliki kegunaan dan keterangan untuk digunakan dalam merancang tabel. Berikut ini beberapa tipe data yang terdapat pada MySQL:

Tabel II.8 Beberapa Tipe Data Pada MySQL

Tipe Data	Keterangan
<i>CHAR</i>	Menampung maksimal M karakter (kombinasi huruf, angka, dan simbol-simbol). Jumlah memori yang dibutuhkan selalu M <i>byte</i> . M terbesar adalah 255.
<i>VARCHAR</i>	Karakter yang disimpan maksimal M karakter. Jumlah memori yang dibutuhkan tergantung jumlah karakter. M bisa mencapai 65535.
<i>DATE</i>	Menyatakan tanggal.
<i>TIME</i>	Menyatakan waktu (jam:menit:detik).
<i>TINYINT</i>	Bilangan antara -128 sampai dengan +127.
<i>SMALLINT</i>	Bilangan antara -32768 sampai dengan +32768.
<i>INT</i>	Bilangan antara -2147683648 sampai dengan +2147683647.
<i>BOOL</i>	Untuk menampung nilai true (benar) dan false (salah). Identik dengan <i>TINYINT</i> .
<i>ENUM</i>	Menyatakan suatu tipe yang nilainya tertentu (disebutkan dalam pendefinisian).
<i>TEXT</i>	Menyimpan teks yang ukurannya sangat panjang.
<i>BLOB</i>	Untuk menyimpan data biner (misalnya gambar atau suara).

(Sumber: Kadir, 2008)

2.17 XAMPP

XAMPP adalah sebuah *software* yang berfungsi untuk menjalankan *website* berbasis PHP dan menggunakan pengolahan data *MySQL* di komputer lokal. *XAMPP* berperan sebagai *web server* pada komputer. *XAMPP* juga dapat disebut sebagai sebuah *CPanel server virtual*, yang dapat membantu melakukan *preview* sehingga dapat memodifikasi *website* tanpa harus *online* atau terakses dengan internet (Kadir, 2008). Bagian yang terpenting dari *XAMPP* adalah sebagai berikut :

1. *Htdoc* adalah *folder* tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas PHP, HTML dan skrip lain.
2. *PhpMyAdmin* merupakan bagian untuk mengelola basis data *MySQL* yang ada di komputer. Untuk membukanya, buka *browser* lalu ketikkan alamat *http://localhost/phpMyAdmin*, maka akan muncul halaman *phpMyAdmin*.
3. Kontrol Panel yang berfungsi untuk mengelola layanan (*service*) *XAMPP*. Seperti menghentikan (*stop*) layanan, ataupun memulai (*start*).

2.18 System Requirement

System requirements atau kebutuhan sistem adalah semua aktivitas yang harus dilakukan atau didukung oleh sistem baru dan batasan-batasan yang harus dicapai sistem baru (Satzinger, 2012). *System requirements* dibagi menjadi 2 kategori, yaitu:

1. *Functional Requirement*

Merupakan penjelasan tentang layanan yang perlu disediakan oleh sistem, bagaimana sistem menerima dan mengolah masukan, dan bagaimana sistem mengatasi situasi-situasi tertentu. Selain itu kadang-kadang juga secara jelas menentukan apa yang tidak dikerjakan oleh sistem. *Functional Requirement* menggambarkan *system requirement* secara detil seperti *input*, *output* dan pengecualian yang berlaku.

Contoh: Sistem mampu menginput dan menampilkan data pelamar kerja.

2. *Non-Functional Requirement*

Secara umum berisi batasan-batasan pada pelayanan atau fungsi yang disediakan oleh sistem. Termasuk di dalamnya adalah batasan waktu, batasan proses pembangunan, standar-standar tertentu. Karena berkaitan dengan kebutuhan sistem secara keseluruhan, maka kegagalan memenuhi kebutuhan jenis ini berakibat pada sistem secara keseluruhan. Contoh: Hak akses admin seleksi penerimaan pegawai dapat melakukan semua kegiatan pada sistem sementara hak akses untuk kepala departemen hanya bisa melihat hasil akhir seleksi.

BAB III

METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Metodologi penelitian adalah tata cara bagaimana suatu penelitian dilaksanakan atau cara yang ditempuh dalam melakukan penelitian. Metodologi penelitian mempelajari tentang langkah-langkah dalam melakukan pengamatan dengan pemikiran yang tepat melalui tahapan-tahapan yang disusun secara sistematis untuk mencari, menyusun, serta menganalisis dan menyimpulkan data berdasarkan fakta-fakta yang ada. Untuk menghasilkan penelitian tugas akhir yang sesuai dengan yang diinginkan maka diperlukan suatu metode dalam penelitian yang telah dipersiapkan sesuai dengan masalah yang akan dibahas.

3.2 Jenis dan Sumber Data

Data yang dibutuhkan dalam penelitian ini dikelompokkan menjadi dua jenis berdasarkan sumbernya, yaitu:

1. Data primer

Data primer merupakan data yang diperoleh langsung dari PT Nusa Indah Jaya Utama, dimana pengumpulan data atau informasi dilakukan langsung dari objek yang diteliti. Data-data tersebut berupa data yang digunakan dalam proses perbaikan dies diantaranya analisis sistem yang telah berjalan serta permasalahan yang terjadi didalam sistem yang berjalan pada perusahaan.

2. Data sekunder

Data sekunder merupakan data yang telah tersedia dan telah terlebih dahulu dikumpulkan dan dihimpun oleh pihak lain. Contoh dari data tersebut adalah data umum PT Nusa Indah Jaya Utama, profil perusahaan dan struktur organisasi perusahaan.

3.3 Metode Pengumpulan Data

Pengumpulan data bertujuan untuk mengumpulkan data dari berbagai sumber yang diperlukan dalam menganalisis dan merancang sistem usulan yang akan dibangun.

Menurut Sugiyono (2013) teknik pengumpulan data merupakan langkah yang paling strategis dalam penelitian, karena tujuan utama dari penelitian adalah mendapatkan data. Penelitian ini menggunakan beberapa teknik pengumpulan data, yaitu:

1. Studi Lapangan

Studi lapangan adalah kegiatan melakukan pengumpulan data secara langsung pada objek yang diteliti, kegiatan tersebut dilakukan dengan cara sebagai berikut:

- a. Pengamatan

Tahap ini dilakukan secara langsung pada divisi *maintenance dies* di PT Nusa Indah Jaya Utama dengan mengamati prosedur perbaikan *dies* yang berjalan, pengamatan dilakukan mulai dari proses diterimanya laporan kerusakan *dies* dari divisi *production* hingga dimana divisi *maintenance dies* mencatat laporan hasil perbaikan *dies* yang telah diperbaiki. Melalui teknik ini, data yang dibutuhkan diamati, dikumpulkan dan diolah sebagai bahan dalam penelitian.

- b. Wawancara

Pengambilan data dengan cara berdialog dan bertanya dengan divisi *maintenance dies*, divisi *production* dan manajer *engineering* tentang proses bisnis perbaikan *dies* yang berjalan.

2. Studi Kepustakaan

Mengumpulkan data dan menambah referensi dengan membaca buku-buku, literatur, artikel di Internet atau sumber tertulis lain yang berhubungan dengan judul dan permasalahan. Studi kepustakaan berguna untuk melengkapi data yang diperlukan dalam penulisan

laporan tugas akhir ini, agar dalam praktik dan teori tidak jauh berbeda.

3.4 Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan adalah metode pengembangan sistem *waterfall*. Metode *waterfall* ini terdiri dari tahap *planning*, *analysis*, *design*, dan *implementation*. Berikut ini akan dijelaskan secara singkat mengenai tahapan-tahapan yang dilakukan dalam pengembangan sistem dengan metode *waterfall*. Berikut ini merupakan tahapan-tahapan metode *waterfall* menurut (Dennis *et al.*, 2015):

1. Tahap Perencanaan (*Planning*)

Pada tahap ini akan dilakukan perencanaan untuk membangun suatu sistem dengan membuat sebuah *system request*.

2. Tahap Analisis (*Analysis*)

Pada tahap ini, merupakan proses analisis kebutuhan sistem. Analisis mengumpulkan data-data sebagai bahan pengembangan sistem. Pengumpulan data dapat dilakukan dengan teknik wawancara ataupun teknik observasi.

3. Tahap Desain (*Design*)

Proses desain pengembang melakukan desain seperti desain struktur data, dan representasi antarmuka. Data-data yang didapat dari tahap analisis diterapkan dalam tahap desain.

4. Tahap Implementasi (*Implementation*)

Pada tahap ini, pengembang akan membangun sebuah sistem dengan rancangan yang sudah ada kemudian diterjemahkan ke dalam sebuah *code* atau aktivitas *coding*. Dimana pada tahap ini *framework* yang digunakan ialah *framework CodeIgniter* dengan menggunakan basis data MySQL.

3.5 Kerangka Penelitian

Dalam penelitian yang dibahas dalam tugas akhir ini, dilakukan langkah-langkah atau tahapan dalam penelitian sesuai dengan metodologi penelitian yang digunakan dalam penelitian tersebut. Tahapan-tahapan dalam penelitian tersebut adalah sebagai berikut:

1. Studi Pendahuluan

Melakukan studi pendahuluan terlebih dahulu dengan mencari dan menentukan judul penelitian serta mencari latar belakang masalah yang terjadi. Selain itu juga membaca buku literatur dan membaca artikel-artikel yang terkait dengan judul penelitian yang akan dibahas.

2. Identifikasi Masalah

Pada tahap ini dilakukan cara dalam mengidentifikasi suatu masalah yang terjadi pada divisi *maintenance dies* dan berfokus pada proses perbaikan *dies*, dalam mengidentifikasi suatu masalah penulis melakukan beberapa cara seperti:

- a. Melakukan wawancara dan observasi yang melibatkan divisi *maintenance dies* dan divisi *production* pada PT Nusa Indah Jaya Utama.
- b. Mengumpulkan data-data yang terkait dalam proses bisnis yang berjalan.

3. Identifikasi Pemecahan Masalah

Setelah mengidentifikasi suatu masalah, maka penulis melakukan pemecahan suatu masalah berdasarkan data-data yang telah di dapat pada tahap sebelumnya. Terdapat beberapa pemecahan masalah yang penulis lakukan seperti:

- a. Menentukan tujuan penelitian serta batasan-batasan pada tugas akhir yang ditulis oleh penulis.
- b. Mengembangkan sistem dengan menggunakan metode *waterfall*, dengan alasan dokumen yang akan diolah akan terorganisir dengan baik. Hal ini dikarenakan setiap fase yang

dilalui harus selesai semua terlebih dahulu sebelum masuk ke fase berikutnya.

4. Identifikasi Kebutuhan Pengguna (*Analysis*)

Pada tahap ini penulis melakukan identifikasi kebutuhan sistem dengan beberapa cara seperti:

- a. Mengumpulkan data-data pada proses bisnis yang berjalan, pengumpulan data-data permasalahan serta menganalisis dokumen masuk dan keluar pada proses perbaikan *dies*.
- b. Identifikasi proses bisnis yang berjalan lalu menganalisis proses bisnis tersebut.
- c. Identifikasi kebutuhan sistem informasi perbaikan *dies*.

5. Desain (*Design*)

Pada tahap ini penulis melakukan perancangan dalam pembuatan sistem dengan menggunakan UML diagram. Dalam pembuatan UML terdapat beberapa diagram seperti:

- a. *Use case diagram* yang bertujuan untuk mendeskripsikan interaksi yang terjadi antara *user* (aktor) dengan sistem yang akan di rancang.
- b. *Use case description* yang bertujuan untuk menjelaskan setiap *use case* yang telah dibuat secara lebih spesifik.
- c. *Activity diagram* yang bertujuan untuk menggambarkan *workflow* atau aliran kerja pada proses bisnis.
- d. *Class diagram* yang bertujuan untuk mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat di antara objek-objek tersebut.
- e. *Sequence diagram* bertujuan untuk menggambarkan perilaku objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang diterima dan dikeluarkan antar objek.
- f. *Deployment diagram* yang bertujuan untuk memvisualisasikan menspesifikasikan serta mendokumentasikan proses yang terjadi pada suatu sistem perangkat lunak yang akan dibangun.

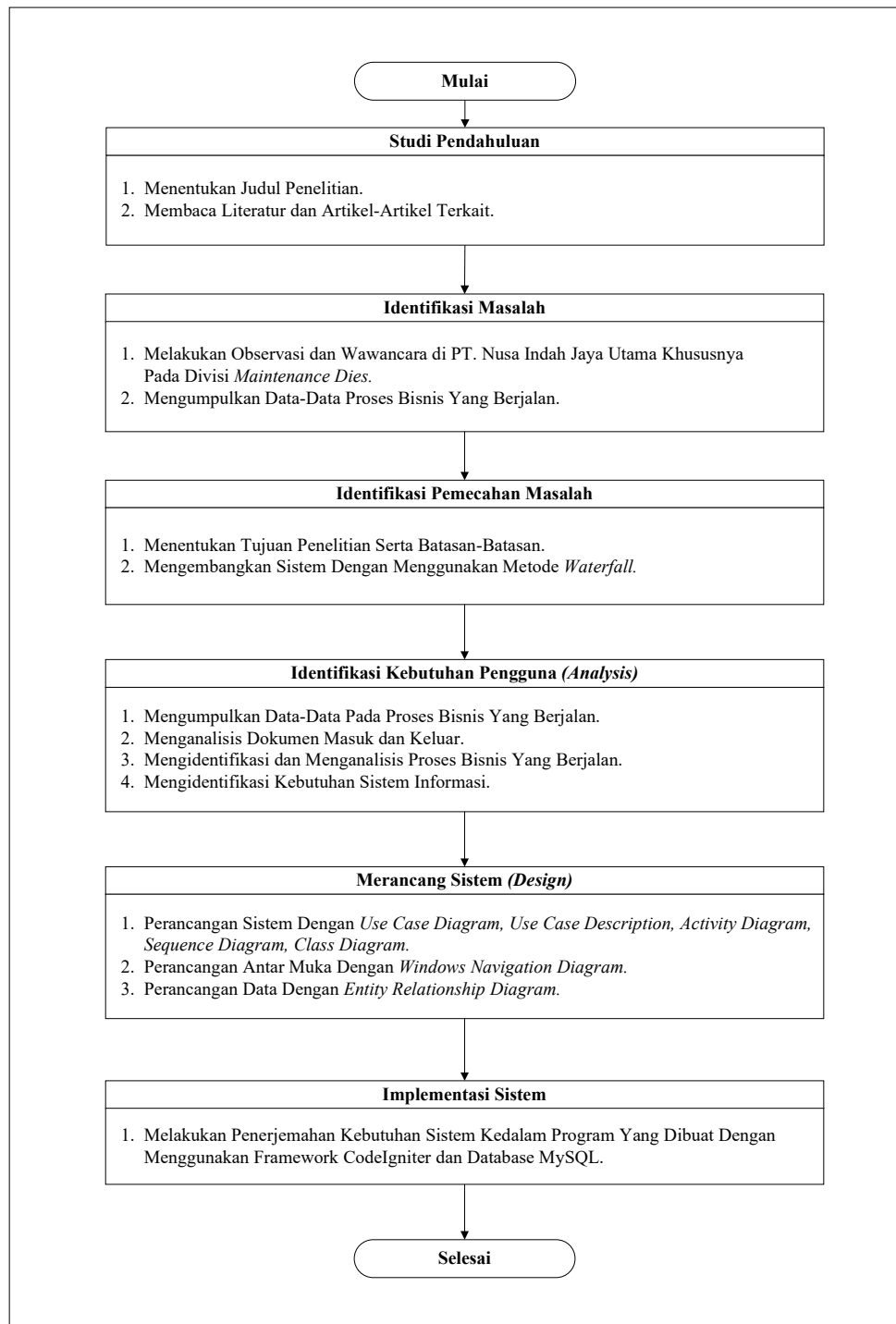
Selain itu dilakukan pembuatan *user interface* dengan metode *windows navigation diagram* (WND) yang bertujuan dalam memberikan suatu gambaran navigasi yang terdapat dalam sistem yang akan dibuat, pembuatan *entity relationship diagram* (ERD) yang bertujuan untuk memodelkan suatu data dan hubungan data serta pembuatan kamus data dengan tujuan untuk memvalidasi diagram alir data dalam hal kelengkapan dan keakuratan data.

6. Implementasi Sistem

Pada tahap terakhir ini penulis menerjemahkannya kedalam sebuah program dengan menggunakan *framework CodeIgniter* dan melakukan pembuatan *database* menggunakan MySQL sesuai dengan analisis yang telah dilakukan pada tahap sebelumnya mengenai proses rekap absensi dan penggajian karyawan.

Berikut adalah kerangka penelitian untuk menyelesaikan masalah dalam Tugas Akhir ini:

.



Gambar III.1 Kerangka Penelitian
(Sumber: Hasil Penelitian, 2019)

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Sejarah Perusahaan

PT Nusa Indah Jaya Utama didirikan oleh bapak H. M. Kusnadi pada tahun 1974 di Jakarta, pada saat itu perusahaan masih terbentuk sebagai CV dengan nama CV Nusa Indah yang bergerak dalam penanganan limbah besi dari kapal -kapal tua. Pada tahun 1976 CV Nusa Indah bekerja sama dengan PT Mitsubishi Kramayudha Motors (PT MKM) dalam penanganan limbah logam industri sisa dari produksi *stamping press*. Dan pada tahun 1978 CV Nusa Indah telah menjadi rekanan tetap PT Mitsubishi Kramayudha Motors khususnya pada bagian penanganan pengerjaan *part* mobil *Colt*.

Pada tahun 1991 tentang kebijakan pemerintah untuk setiap industri otomotif yang berada di Indonesia khususnya penanaman modal asing diwajibkan memberi *support* kepada industri dalam negeri untuk memajukan serta mengembangkan perusahaannya agar komponen lokal yang digunakan dalam produknya bisa memiliki kualitas yang baik. Dan dari sebab itu CV Nusa Indah diberikan pekerjaan oleh PT Mitsubishi Kramayudhan Motors (PT MKM) untuk memproduksi *clip hardness* menggunakan mesin *portable press* dengan ukuran 25 ton, 40 ton, 60 ton, dan 80 ton. Pada tahun yang sama CV Nusa Indah berpindah lokasi ke Jl. Pulogebang dan menempati lahan dengan luas area industri yaitu 1.350 m². Pada tahun 2007 CV Nusa Indah berubah bentuk menjadi Perseroan Terbatas (PT) dan merubah nama menjadi PT Nusa Indah Jaya Utama sekaligus berpindah alamat di Jl. Laskar 49 Pekayon, Bekasi Selatan

Pada tahun 2013 PT Nusa Indah Jaya Utama mulai menggunakan mesin *medium press* yang didatangkan dari China dan Korea dalam mengerjakan pesanan. Pada tahun 2014 PT Nusa Indah Jaya Utama membeli mesin *high press* dengan kapasitas 200 sampai dengan 315 ton untuk menjadi *vendor* PT Gemala Kempa Daya, PT Fuji Teknika Indonesia, dan PT Inti Pantja Press Industri.

4.2 Profil Perusahaan

Nama Perusahaan	: PT Nusa Indah Jaya Utama
Lokasi	: Jl. Laskar Raya No. 49 RT. 003 RW. 002 Kel. Pekayon Jaya, Kec. Bekasi Selatan
Telepon	: 021-82411782 / 021 -8201008
Fax	: 021-82411782
Direktur Utama	: Bpk. H. M. Kusnadi
Email	: ptniju@gmail.com
Aktivitas Bisnis	: <i>Stamping dan Manufacturing</i>
Status	: Perseroan Terbatas
Produksi	: Komponen Otomotif
Perizinan	: <ul style="list-style-type: none"> • SIUP: 510/656-BPPT/PM/IX/2013 • NPWP No: 02.182.710.0-006.000 • TDP: 102614611073

Logo dari PT Nusa Indah Jaya Utama dapat dilihat pada Gambar III.1 berikut:



Gambar IV.1 Logo PT Nusa Indah Jaya Utama
(Sumber: PT Nusa Indah Jaya Utama, 2018)

4.3 Visi dan Misi Perusahaan

Pengertian visi berdasarkan istilah yaitu suatu pandangan jauh tentang suatu perusahaan atau lembaga lainnya, baik itu tentang tujuan - tujuan perusahaan, maupun apa yang harus dilakukan perusahaan untuk mencapai tujuan tersebut pada masa yang akan datang.

Sedangkan pengertian misi ialah suatu pernyataan tentang apa yang harus dikerjakan oleh perusahaan atau lembaga dalam usaha mewujudkan visi tersebut. Misi perusahaan diartikan sebagai tujuan dan alasan mengapa perusahaan atau lembaga itu dibuat. Misi juga akan memberikan arah sekaligus batasan-batasan proses pencapaian tujuan.

4.3.1 Visi

Menjadi perusahaan penyedia jasa *stamping* dan *manufacturing* untuk komponen otomotif, baik roda 2 maupun roda 4.

4.3.2 Misi

Membangun perusahaan yang mampu menciptakan lapangan kerja berkualitas bagi sebanyak mungkin rakyat Indonesia dengan mengusung nilai-nilai berikut ini:

1. Pengembangan kompetensi karyawan secara berkelanjutan
2. Mengupayakan pertumbuhan finansial, intelektual dan citra perusahaan yang konsisten serta melakukan investasi kembali ke dalam bisnis yang dijalankan, dan
3. Mempertahankan standar kode etik yang tinggi dalam aktivitas bisnis.

4.4 Struktur Organisasi

Struktur Organisasi ialah merupakan suatu susunan dan hubungan antara tiap bagian atau posisi yang ada pada suatu organisasi ataupun perusahaan dalam menjalankan kegiatan operasional untuk mencapai tujuan yang di harapkan dan diinginkan. Struktur organisasi menggambarkan dengan jelas pemisahan kegiatan pekerjaan antara kegiatan yang satu dengan kegiatan yang lain dan bagaimana

hubungan aktivitas dan fungsi dibatasi. Dalam struktur organisasi yang baik harus menjelaskan hubungan wewenang serta pertanggung jawaban apa yang akan dikerjakan.

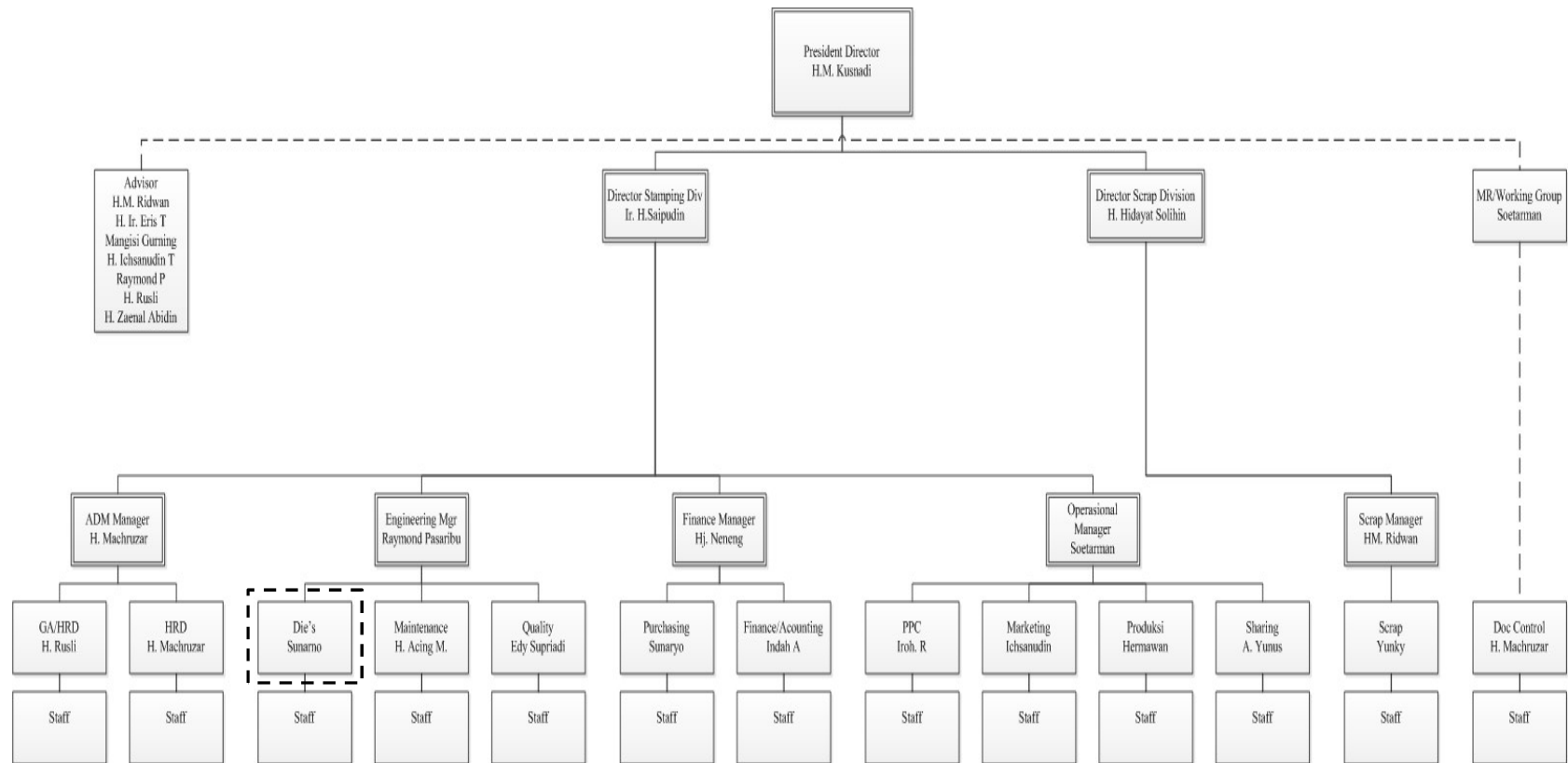
Dengan adanya struktur organisasi dapat memberikan informasi kepada seluruh anggota atau karyawan perusahaan tentang kegiatan yang harus dikerjakan serta pertanggung jawaban kegiatan diberikan kepada pihak mana, sehingga proses kerjasama antara bagian satu dengan bagian yang lainnya bisa saling bersinergi dan dapat mencapai tujuan perusahaan dengan cepat dan baik.

Struktur organisasi yang terdapat pada PT Nusa Indah Jaya Utama ialah struktur organisasi lini/garis. PT Nusa Indah Jaya Utama menggunakan struktur organisasi ini dikarenakan pada struktur organisasi ini perintah terpusat pada atasan secara langsung serta koordinasi dengan bawahannya secara langsung, sehingga perintah dapat diterima dengan jelas dan mudah dilaksanakan berdasarkan tanggung jawabnya masing-masing.

Pada struktur organisasi lini/garis ini, wewenang dari atasan disalurkan secara vertikal kepada bawahan, begitu juga sebaliknya. Pertanggung jawaban dari bawahan secara langsung ditujukan kepada atasan yang memberi perintah.

Ciri - ciri struktur organisasi lini/garis:

1. Mempunyai wewenang dan tanggung jawab secara langsung secara vertikal yang dikaitkan dengan jabatan dan tugas tiap tingkatan.
2. Bawahan hanya mempunyai satu atasan.
3. Adanya kesatuan komando atau perintah.
4. Hanya cocok untuk organisasi yang kecil atau sederhana.



Ket :



Tempat penelitian tugas akhir berlangsung

Gambar IV.2 Struktur Organisasi PT Nusa Indah Jaya Utama

(Sumber: PT Nusa Indah Jaya Utama, 2018)

4.5 *Job Description*

Job Description merupakan sebuah kumpulan informasi jabatan dan disusun secara sistematis yang dapat mengidentifikasi dan menguraikan suatu jabatan atau posisi tertentu. *Job description* membuat status setiap jabatan menjadi jelas akan fungsi serta tanggung jawabnya. Dari struktur organisasi di atas, maka *job description* dari masing-masing divisi adalah sebagai berikut:

1. *President Director* bertugas dalam hal:
 - a. Bertanggung jawab dalam pengambilan keputusan serta kebijakan yang diambil oleh perusahaan.
 - b. Berkoordinasi dengan *director*, *manager* serta semua kepala divisi untuk menentukan keputusan serta kebijakan yang akan diambil oleh perusahaan.
2. *Director* bertugas dalam hal:
 - a. Bertanggung jawab kepada *President Director* dalam setiap kegiatan yang dilakukan oleh perusahaan.
 - b. Menentukan kebijakan dan mengambil keputusan terhadap manajemen perusahaan.
 - c. Menentukan struktur organisasi pada perusahaan serta operasional perusahaan.
3. *Administration Manager* bertugas dalam hal:
 - a. Bertanggung jawab kepada *Director* terhadap kegiatan administrasi yang terjadi di perusahaan.
 - b. Membuat perencanaan strategis operasional perusahaan berkaitan dengan sumberdaya.
 - c. Memonitor dan melengkapi kebutuhan sumberdaya manusia.
 - d. Melakukan pengawasan dan realisasi proses rekrutmen karyawan.
4. *Engineering Manager* bertugas dalam hal:
 - a. Bertanggung jawab kepada *Director* terhadap mesin yang terdapat pada perusahaan.

- b. Memonitor terhadap pencapaian kualitas yang dibuat oleh departemen produksi.
 - c. Membuat laporan setiap bulannya mengenai kualitas produk, mesin dan *dies*.
- 5. *Finance Manager* bertugas dalam hal:
 - a. Bertanggung jawab kepada *Director* terhadap keuangan perusahaan.
 - b. Memonitor pelaksanaan pembayaran ke *supplier* dan penagihan ke pelanggan.
 - c. Memonitor terhadap proses seleksi dan evaluasi *supplier*.
 - d. Membuat laporan bulanan tentang seleksi dan evaluasi *supplier* dan laporan keuangan perusahaan.
- 6. *Operational Manager* bertugas dalam hal:
 - a. Bertanggung jawab kepada *Director* terhadap kegiatan operasional yang terjadi di perusahaan.
 - b. Membuat perencanaan strategis operasional perusahaan.
 - c. Memonitor terhadap pencapaian pengiriman, produksi serta penjualan.
- 7. *Scrap Manager* bertugas dalam hal:
 - a. Bertanggung jawab kepada *Director* terhadap pengumpulan serta penjualan *scrap*.
 - b. Menentukan *layout* dari penyimpanan *scrap* yang akan disimpan di gudang.
 - c. Membuat laporan hasil pengumpulan serta penjualan *scrap* setiap bulannya.
- 8. *Human Resource Development* bertugas dalam hal:
 - a. Bertanggung jawab kepada *Administration Manager* terhadap kinerja karyawan yang ada pada perusahaan.
 - b. Melakukan sosialisasi terhadap karyawan tentang peraturan yang ada pada perusahaan.
 - c. Membuat laporan penilaian kinerja karyawan.

9. *Health, Safety, Environment* bertugas dalam hal:
 - a. Bertanggung jawab kepada *Administration Manager* terhadap keselamatan dan kesehatan kerja dalam perusahaan.
 - b. Memberikan pelatihan terhadap karyawan tentang antisipasi kecelakaan termasuk kebakaran.
 - c. Melakukan perawatan secara berkala terhadap sarana keselamatan dan kesehatan kerja.
10. *Maintenance Dies* bertugas dalam hal:
 - a. Bertanggung jawab kepada *Engineering Manager* terhadap perawatan serta perbaikan *dies* yang terdapat pada perusahaan.
 - b. Membuat *list dies* yang digunakan untuk proses produksi.
 - c. Membuat perencanaan perawatan *dies* secara berkala.
 - d. Melakukan perawatan *dies* secara berkala.
 - e. Melakukan perbaikan *dies* apabila terjadi kerusakan.
 - f. Melaporkan hasil perawatan *dies* kepada *Engineering Manager*.
 - g. Melaporkan hasil perbaikan *dies* kepada *Engineering Manager*.
11. *Maintenance Machine* bertugas dalam hal:
 - a. Bertanggung jawab kepada *Engineering Manager* terhadap perawatan serta perbaikan mesin yang terdapat pada perusahaan.
 - b. Membuat *list* mesin yang digunakan dalam proses produksi.
 - c. Membuat perencanaan perawatan mesin dan alat secara berkala.
 - d. Melakukan perawatan mesin dan alat secara berkala.
 - e. Melakukan perbaikan mesin apabila terjadi kerusakan.
 - f. Melaporkan hasil perawatan mesin kepada *Engineering Manager*.
 - g. Melaporkan hasil perbaikan mesin kepada *Engineering Manager*.
12. *Quality Control* bertugas dalam hal:
 - a. Bertanggung jawab kepada *Engineering Manager* terhadap kualitas produk yang dihasilkan oleh perusahaan.
 - b. Melakukan identifikasi terhadap masalah kualitas produk dalam tahap *Incoming, Inprocess & Out Going*

- c. Melaporkan hasil temuan kualitas produk baik dalam tahap *Incoming, Inprocess & Out Going* kepada *Engineering Manager*.
13. *Purchasing* bertugas dalam hal:
- a. Bertanggung jawab kepada *Finance Manager* terhadap pembelian barang yang dilakukan oleh perusahaan.
 - b. Membuat perencanaan dan melakukan proses seleksi terhadap *supplier* baru sebelum melakukan pembelian barang.
 - c. Membuat evaluasi terhadap *supplier* yang telah ada.
 - d. Membuat laporan pembelian barang yang dilakukan oleh perusahaan.
14. *Finance/Accounting* bertugas dalam hal:
- a. Bertanggung jawab kepada *Finance Manager* terhadap keuangan perusahaan.
 - b. Membuat perencanaan dan melakukan proses pembayaran kepada *supplier*.
 - c. Membuat perencanaan dan melakukan proses penagihan pembayaran kepada pelanggan.
 - d. Membuat rekapitulasi dan perhitungan pajak.
 - e. Membuat laporan keuangan setiap bulan.
15. *Production Planning Control* bertugas dalam hal:
- a. Bertanggung jawab kepada *Operational Manager* terhadap perencanaan produksi yang akan dilakukan perusahaan.
 - b. Mengolah informasi *order* dari *marketing* menjadi *planning* produksi dan *delivery*.
 - c. Mendistribusikan *planning* produksi kepada divisi produksi.
 - d. Menyiapkan bahan baku untuk kebutuhan proses produksi.
 - e. Melakukan monitoring pencapaian proses produksi.
 - f. Membuat laporan pencapaian produksi perusahaan.
16. *Marketing* bertugas dalam hal:
- a. Bertanggung jawab kepada *Operational Manager*, terhadap penawaran produk yang dihasilkan perusahaan kepada pelanggan.

- b. Melakukan komunikasi dengan pelanggan untuk mendapatkan *order*.
 - c. Membuat laporan evaluasi tentang kepuasan pelanggan kepada perusahaan.
- 17. *Production* bertugas dalam hal:
 - a. Bertanggung jawab kepada *Operational Manager* terhadap proses produksi yang terjadi pada perusahaan.
 - b. Membuat laporan hasil proses produksi sesuai dengan hasil produksi.
 - c. Membuat laporan masalah jika dalam proses produksi terjadi masalah.
- 18. *Shearing* bertugas dalam hal:
 - a. Bertanggung jawab kepada *Operational Manager* terhadap proses pemotongan lembaran metal yang terjadi pada perusahaan.
 - b. Membuat laporan hasil dari proses pemotongan lembaran metal yang telah ditentukan ukurannya.
- 19. *Scrap* bertugas dalam hal:
 - a. Bertanggung jawab kepada *Scrap Manager* terhadap pengumpulan dan penyimpanan *scrap*.
 - b. Melakukan pengumpulan serta penyimpanan *scrap* agar sesuai dengan *layout* yang telah ditentukan.

4.6 Tenaga Kerja

Tenaga kerja ialah merupakan orang-orang yang terlibat langsung di dalam proses produksi, menggunakan tenaga maupun pikiran mereka dalam menjalankan proses produksi. Oleh karena itu, tenaga kerja dapat dikatakan sebagai salah satu faktor penting didalam perusahaan. Tanpa adanya tenaga kerja, kegiatan produksi di suatu perusahaan tidak akan berjalan. Ditinjau dari aktivitas yang dilakukan, berdasarkan pekerjaannya tenaga kerja dapat dibagi menjadi tiga, yaitu:

1. Tenaga Kerja Perencana

Yaitu mereka yang memiliki keahlian untuk menyusun dan merumuskan perencanaan yang diperlukan perusahaan kedepannya baik itu dalam bagian *Operational, Finance, Engineering* maupun *Administration*. Contohnya adalah *Manager* yang terdapat pada bagian yang telah ditentukan.

2. Tenaga Kerja Pelaksana

Yaitu mereka yang secara langsung melaksanakan aktivitas yang sudah direncanakan baik dalam masalah produksi, pemasaran, maupun administrasi. Contohnya adalah *Staff Operational* yang terdapat pada bagian yang telah di tentukan.

3. Tenaga Pengawas

Yaitu mereka yang bertugas melakukan pengawasan terhadap aktivitas dari pekerja pelaksana dan memberikan masukan apabila dibutuhkan. Contohnya adalah *Leader* yang terdapat pada bagian yang telah di tentukan.

4.7 Lokasi Perusahaan

Lokasi perusahaan merupakan tempat dimana perusahaan melakukan aktivitasnya. Lokasi perusahaan juga dapat menentukan maju atau tidaknya sebuah perusahaan. Dengan lokasi yang mudah dijangkau dan strategis, maka akan memudahkan perusahaan untuk mencapai tujuan dan target pasar yang akan dituju oleh perusahaan.



Adapun mengenai lokasi perusahaan pada PT Nusa Indah Jaya Utama di antaranya sebagai berikut:

1. Pabrik untuk produksi cukup luas sehingga pekerja bisa leluasa dalam melakukan pekerjaan.
2. Mudah dan cepat untuk mendapatkan bahan baku karena lokasi pemasok yang dekat dengan lokasi perusahaan.
3. Lokasi sangat strategis yaitu dekat dengan jalan besar dan jalan bebas hambatan (jalan tol).

4.8 Produk Perusahaan

PT Nusa Indah Jaya Utama memproduksi beberapa jenis produk *spare part* kendaraan baik kendaraan roda dua maupun kendaraan roda empat, berikut ini beberapa produk yang diproduksi PT Nusa Indah Jaya Utama diantaranya adalah:

Tabel IV.1 Produk Perusahaan PT Nusa Indah Jaya Utama

No.	Nama Produk	Proses	Mesin	Gambar
1	<i>Pan Frame Rear Back RH</i>	<i>Drawing Trimming + Piercing Flange + Burring</i>	315 Ton 200 Ton 200 Ton	
2	<i>Pan Frame Rear Back LH</i>	<i>Drawing Trimming + Piercing Flange + Burring</i>	250 Ton 160 Ton 200 Ton	
3	<i>Bracket Brake Pedal</i>	<i>Drawing Trimming Piercing</i>	315 Ton 160 Ton 160/110 Ton	
4	<i>Bracket Clutch Pedal</i>	<i>Drawing Trimming</i>	315 Ton 160 Ton	

(Sumber: PT Nusa Indah Jaya Utama, 2018)

Tabel IV.1 Produk Perusahaan PT Nusa Indah Jaya Utama (lanjutan)




No.	Nama Produk	Proses	Mesin	Gambar
5	<i>Rear Bracket Bottom</i>	<i>Blank Forming Piercing</i>	160 Ton 160 Ton 160 Ton	
6	<i>Utility Mtg Bracket Front</i>	<i>Blank Bending Forming</i>	160 Ton 160 Ton 160 Ton	
7	<i>Bridge Tube Upper Middle</i>	<i>Blank Forming Piercing</i>	160 Ton 160 Ton 40 Ton	
8	<i>Bkt Cover Main Tube Mtg</i>	<i>Blank Bending</i>	110 Ton 110 Ton	
9	<i>Sealing Cowl Ventilator</i>	<i>Drawing Trimming</i>	315 Ton 160 Ton	

(Sumber: PT Nusa Indah Jaya Utama, 2018)

4.9 Dies




Dies ialah suatu cetakan yang digerakan oleh mesin *press* untuk menekan bahan atau *material* agar menghasilkan barang yang sesuai dengan keinginan. Berikut beberapa *dies* yang dimiliki oleh PT Nusa Indah Jaya Utama :

Tabel IV.2 Dies PT Nusa Indah Jaya Utama

<i>No</i>	<i>Part Name</i>	Kode / Proses ke	<i>Customer</i>	Gambar
1	<i>Bridge Tube Upper Middle</i>	SGS 53 / 3 of 3 (Piercing)	PT Setia Guna Selaras	
2	<i>Rear Bracket Bottom</i>	SGS 49 / 1 of 3 (Blank)	PT Setia Guna Selaras	
3	<i>Pan Frame Rear Back RH</i>	PF RH / 2 of 3 (Trim + Piercing)	PT Posmi Steel Indonesia	

(Sumber: PT Nusa Indah Jaya Utama, 2018)

Tabel IV.2 Dies PT Nusa Indah Jaya Utama (lanjutan)




No	Part Name	Kode / Proses ke	Customer	Gambar
4	<i>Pan Frame Rear Back RH</i>	PF RH / 1 of 3 (Drawing)	PT Posmi Steel Indonesia	
5	<i>Bonnet Upper Fuel Tank</i>	BONET UPR / 4 of 4 (Piercing)	PT Karya Linggom Prima	
6	<i>Bonnet Upper Fuel Tank</i>	BONET UPR / 4 of 4 (Trim)	PT Karya Linggom Prima	

(Sumber: PT Nusa Indah Jaya Utama, 2018)

4.10 Alat-Alat Yang Digunakan Dalam Proses Perbaikan




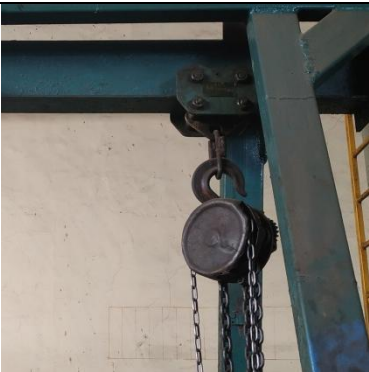
Dalam melakukan proses perbaikan dies, divisi maintenance dies memiliki beberapa alat yang biasa digunakan untuk membantu proses kegiatan dan perbaikan. Berikut peralatan yang biasa digunakan:

Tabel IV.3 Peralatan *Maintenance Dies* PT Nusa Indah Jaya Utama

No	Nama Alat	Spesifikasi	Gambar	Deskripsi
1	Mesin <i>Grinding</i>	70 x 40 mm		Digunakan untuk mengasah permukaan dies.
2	Mesin Las	250 NL		Digunakan untuk mengelas bagian dies.
3	Mesin Potong	ZAY 7045FG		Digunakan untuk memotong bagian dari dies.



(Sumber: PT Nusa Indah Jaya Utama, 2018)

Tabel IV.3 Peralatan *Maintenance Dies* PT Nusa Indah Jaya Utama (lanjutan)

No	Nama Alat	Spesifikasi	Gambar	Deskripsi
4	Meja Repair (1)	1300 x 1300 mm		Digunakan sebagai tempat untuk memperbaiki <i>dies</i> .
5	Meja Repair (2)	1600 x 850 mm		Digunakan sebagai tempat untuk memperbaiki <i>dies</i> .
6	Crane (1)	3 Ton		Digunakan untuk mengangkat <i>dies</i> yang memiliki beban ≤ 3 Ton
7	Crane (2)	1,5 Ton		Digunakan untuk mengangkat <i>dies</i> yang memiliki beban $\leq 1,5$ Ton

(Sumber: PT Nusa Indah Jaya Utama, 2018)

Tabel IV.3 Peralatan *Maintenance Dies* PT Nusa Indah Jaya Utama (lanjutan)

No	Nama Alat	Spesifikasi	Gambar	Deskripsi
8	Gerinda	Gerinda Angin		Digunakan untuk mengasah permukaan <i>dies</i> .
9	Gerinda Lurus	Gerinda Angin		Digunakan untuk mengasah permukaan <i>dies</i> .

(Sumber: PT Nusa Indah Jaya Utama, 2018)

4.11 Job Description Divisi *Maintenance Dies*

Divisi *maintenance dies* memiliki beberapa *job desc* diantaranya yaitu:

1. Bertanggung jawab kepada *Engineering Manager* terhadap perawatan serta perbaikan *dies* yang terdapat pada perusahaan.
2. Membuat *list dies* yang digunakan untuk proses produksi.
3. Melaporkan hasil perawatan dan perbaikan *dies* kepada *Engineering Manager*.

4.11.1 Proses Perbaikan *Dies*

Proses perbaikan *dies* pada PT Nusa Indah Jaya Utama antara lain adalah sebagai berikut:

1. Pada proses perbaikan *dies* dimulai dengan divisi *production* membuat laporan kerusakan *dies* pada form *maintenance dies job order*.
2. Setelah selesai dibuat divisi *production* menyerahkan form tersebut kepada divisi *maintenance dies* untuk dilakukan analisis kerusakan yang terjadi pada *dies* tersebut.
3. Setelah selesai menganalisis kerusakan, divisi *maintenance dies* akan memberikan keputusan apakah *dies* dalam kondisi rusak parah dan membutuhkan penanganan khusus atau tidak.
4. Apabila *dies* dalam kondisi rusak parah dan membutuhkan penanganan khusus, maka divisi *maintenance dies* akan meminta persetujuan kepada *customer* untuk memperbaiki *dies* kepada rekanan PT Nusa Indah Jaya Utama.
5. Jika *customer* menyetujui, maka divisi *maintenance dies* akan membawa *dies* ke *workshop* rekanan, namun jika *customer* tidak menyetujui maka *dies* akan diperbaiki oleh pihak *customer*.
6. Namun apabila kondisi *dies* tidak membutuhkan penanganan khusus, maka divisi *maintenance dies* akan melakukan perbaikan sendiri pada PT Nusa Indah Jaya Utama.
7. Setelah selesai melakukan perbaikan *dies*, divisi *maintenance dies* akan melakukan peng-ngecekan kelayakan *dies* baik itu dilakukan perbaikan pada divisi *maintenance dies*, *workshop* rekanan maupun diperbaiki oleh *customer*. Apabila *dies* masih belum layak, maka *dies* akan dilakukan perbaikan lagi hingga kondisi *dies* layak digunakan.
8. Namun apabila kondisi *dies* sudah layak untuk digunakan, maka divisi *maintenance dies* akan mencatat hasil perbaikan kedalam buku besar yang nantinya akan dilaporkan kepada engineering manager.


(Sumber: Hasil Analisis, 2019)

4.12 Dokumen Terkait Proses Perbaikan *Dies*

Dalam melakukan proses perbaikan *dies* pastilah terdapat dokumen yang terlibat dalam proses tersebut, dokumen-dokumen yang terlibat dalam proses tersebut digunakan sebagai sumber informasi dari suatu proses yang akan dikerjakan dalam proses perbaikan *dies*. Berikut merupakan dokumen-dokumen yang terlibat dalam proses tersebut.

1. *Maintenance Dies Job Order*

Merupakan form yang digunakan untuk mencatat apabila terdapat kerusakan pada *dies*. Form *Maintenance Dies Job Order* dapat dilihat pada Gambar IV.4 berikut:

	PT. NUSA INDAH JAYA UTAMA	Nomor : FR-NUJ-MCE-01-06
	MAINTENANCE DIES JOB ORDER	Revisi : 0
		Tgl. Berlaku : 31 - 08 - 2017
		Halaman :
No : MCE JO/ _____ Tanggal : _____		KEPADA : DIVISI MAINTENANCE DIES
NAMA DIES : _____ NO PROSES : _____ NAMA PROSES : _____		
URAIAN KERUSAKAN : _____ _____ _____ _____		
<div style="border: 1px solid black; padding: 2px; text-align: center;">DI ISI OLEH MAINTENANCE</div>		
TANGGAL PERBAIKAN : _____ URAIAN PERBAIKAN : _____ _____ _____ _____		

Gambar IV.4 Form *Maintenance Dies Job Order*
(Sumber: PT Nusa Indah Jaya Utama, 2018)

Keterangan:

Asal : Divisi *Production*
 Tujuan : Divisi *Maintenance Dies*
 Frekuensi : Dokumen ini diterbitkan apabila terjadi kerusakan pada *dies*.

Kolom-kolom	:	<ul style="list-style-type: none"> No 	:	Nomer dokumen
		<ul style="list-style-type: none"> Tanggal 	:	Tanggal pengisian dokumen
		<ul style="list-style-type: none"> Kepada 	:	Tujuan pengiriman dokumen
		<ul style="list-style-type: none"> Nama <i>Dies</i> 	:	Nama <i>dies</i> yang rusak
		<ul style="list-style-type: none"> No Proses 	:	Nomer proses
		<ul style="list-style-type: none"> Nama Proses 	:	Nama proses
		<ul style="list-style-type: none"> Uraian Kerusakan 	:	Uraian kerusakan yang terjadi pada <i>dies</i>
		<ul style="list-style-type: none"> Tanggal Perbaikan 	:	Tanggal perbaikan <i>dies</i>
		<ul style="list-style-type: none"> Uraian Perbaikan 	:	Uraian kegiatan perbaikan yang dilakukan
		<ul style="list-style-type: none"> Diketahui Oleh 	:	Mengetahui engineering manager

2. *List Dies*

Merupakan yang digunakan untuk mencatat dies yang terdapat pada divisi *maintenance dies*. *List dies* dapat dilihat pada Gambar IV.5 berikut:

[illegible]

Gambar IV.5 Form *List Dies*
(Sumber: PT Nusa Indah Jaya Utama, 2018)

Keterangan:

- Asal : Divisi *Maintenance Dies*
- Tujuan : Divisi *Maintenance Dies*
- Frekuensi : Dokumen ini diterbitkan untuk mencatat *dies* apa saja yang terdapat pada perusahaan.
- Kolom-kolom : • No : Nomer urut
- Nama *Dies* : Nama *dies*
- No *Dies* : Nomer Proses
- Nama Proses: Nama Proses Pekerjaan
- *Customer* : Nama *Customer*

3. Buku Besar Pencatatan Hasil Perbaikan

Merupakan buku yang digunakan untuk mengevaluasi hasil dari perbaikan yang dilakukan. Buku besar pencatatan hasil perbaikan dapat dilihat pada Gambar IV.6 berikut:

No.	DATE (TGL)	NAME DIE	DIES PROBLEM	DIES REPAIR
		AGUSTUS		
1	22-08-2017	KLIP 60 PROSESS	HAJI BAHU	DIES BLOKING DI GRINDING
2	24-08-2017	KLIP 60 PROSESS	BARRY	OK
3	24-08-2017	SGS 30 TRIM	BART BROS	OK
4	25-08-2017	KLIP 40 PROSESS	HADI BUNG	DIES BLOK DI GRINDING
5	30-08-2017	771 44/45 3/4	BARTY	OK
6	31-08-2017	CUPA 017	HASIL BANG	OK
			RUEP.	PATAH TELAH DISANT
				SUDAH DI GRINDING
				DIES ATAS DIBANDING
				SPRING PATAH BAWAH. TUTUKU
				SPRING.
		SEPTEMBER		
1	04-09-2017	SGS 40 96	HADI BANG	
2	04-09-2017	CUP. B BANG	BARTY	GRINDING DIES
				ATAS dan dies bawah
				OK. SETTING LAGI
3	05-09-2017	CUP. A. BANG	BANG BARTY	TAGGAL PROD.
				GRINDING DIES ATAS
				dan DIES BAWAH
				SETTING OK.
4	06-09-2017	KLIP 90 PROSESS	BARTY	DIE DIES ATAS BANG
				DI GRINDING SET OK.
5	07-09-2017	KLIP 60 PROSESS	BANG BARTY	DIES TROK ATAS dan BAWAH
				di grinding dan
				SETTING OK.
6	08-09-2017	KLIP 60 PROSESS		DIES BLOKING di grinding
				OK.
7	10-09-2017	G. 38 TRIM	PLAT HOSIANG	GRINDING OK.
			BARTY	
8	11-09-2017	G. 38 TRIM	PLAT HOSIANG	GRINDING OK.
			BARTY	
9	14-09-2017	SGS 30 PROSESS	HASIL BANG	GRINDING OK.
10	17-09-2017	SGS 30 PROSESS	PLAT BANG	GRINDING OK.

Gambar IV.6 Buku Besar Pencatatan Hasil Perbaikan
(Sumber: PT Nusa Indah Jaya Utama, 2018)

Keterangan:

- Asal : Divisi *Maintenance Dies*
- Tujuan : Divisi *Maintenance Dies*
- Frekuensi : Buku ini digunakan untuk mencatat hasil perbaikan yang telah dilakukan oleh divisi *maintenance dies*.
- Kolom-kolom : • Tanggal : Tanggal perbaikan
 • Nama *Dies* : Nama *dies* yang terjadi kerusakan
 • *Dies Problem* : Kerusakan *dies*
 • *Dies Repair* : Perbaikan *dies* yang dilakukan

4.13 Pengolahan Data Kerusakan *Dies*

Penilaian kerusakan dies pada PT Nusa Indah Jaya Utama bersifat statis, dimana bobot dari kriteria penilaian tidak dapat diubah dengan semauanya dikarenakan part dari dies memiliki tingkat kepentingannya tersendiri. Pada proses penilaian kerusakan dies memiliki beberapa kriteria seperti dies plat, guide post, spring, die dan punch.

4.13.1 Kriteria dan Subkriteria

Dalam penilaian kerusakan *dies* terdapat beberapa kriteria dan juga sub-kriteria, diantaranya yaitu sebagai berikut :

Tabel IV.4 Kriteria Penilaian Kerusakan *Dies*

Kriteria	Bobot
<i>Dies Plate (K1)</i>	60
<i>Guide Post (K2)</i>	15
<i>Spring (K3)</i>	10
<i>Punch (K4)</i>	8
<i>Die (K5)</i>	7
Total	100

(Sumber: Hasil Analisis, 2019)

Adapun untuk sub kriteria dari penilaian kerusakan dies yaitu sebagai berikut:

Tabel IV.5 Subkriteria Penilaian Kerusakan Dies

Kriteria	Parameter	Nilai Parameter
K1	Pecah	3
	Retak	2
	Tidak Pecah	1
K2	Rusak	2
	Tidak Rusak	1
K3	Rusak	2
	Tidak Rusak	1
K4	Rusak	2
	Tidak Rusak	1
K5	Rusak	2
	Tidak Rusak	1

(Sumber: Hasil Analisis, 2019)

4.13.2 Contoh Kasus

Berikut merupakan contoh kasus dari penerapan metode *Simple Multi Attribute Rating Technique* (SMART).

Tabel IV. 6 Alternatif

No	Alternatif / Nama Dies
1	<i>Dies</i> SGS (A1)
2	<i>Dies</i> Posmi (A2)
3	<i>Dies</i> KLP (A3)
4	<i>Dies</i> MKM (A4)

(Sumber: Hasil Analisis, 2019)

Langkah pertama sebelum melakukan perhitungan dengan metode smart ialah dengan menentukan alternatif apa saja yang akan dimasukkan kedalam perhitungan

Tabel IV.7 Kriteria

No	Kriteria	Bobot
1	Dies Plate (K1)	60
2	Guide Post (K2)	15
3	Spring (K3)	10
4	Punch (K4)	8
5	Die (K5)	7
Total		100

(Sumber: Hasil Analisis, 2019)

Setelah menentukan alternatif dilanjutkan dengan menentukan kriteria beserta bobotnya untuk dilakukan perhitungan.

Tabel IV.8 Nilai Parameter

No	Kriteria (c)	Parameter	Nilai Parameter
1	K1	Tidak Pecah	1
		Retak	2
		Pecah	3
2	K2	Tidak Rusak	1
		Rusak	2
3	K3	Tidak Rusak	1
		Rusak	2
4	K4	Tidak Rusak	1
		Rusak	2
5	K5	Tidak Rusak	1
		Rusak	2

(Sumber: Hasil Analisis, 2019)

Setelah menentukan kriteria beserta bobotnya dilanjutkan dengan menentukan nilai parameter dari setiap sub kriteria yang ada.

Tabel IV. 9 Kriteria Dari Setiap Alternatif

Alternatif	Kriteria				
	(K1)	(K2)	(K3)	(K4)	(K5)
A1	Pecah	Tidak Rusak	Rusak	Tidak Rusak	Rusak
A2	Tidak Pecah	Rusak	Tidak Rusak	Rusak	Tidak Rusak
A3	Pecah	Rusak	Rusak	Rusak	Rusak
A4	Tidak Pecah	Rusak	Rusak	Rusak	Rusak

(Sumber: Hasil Analisis, 2019)

Parameter yang keluar dari setiap kriteria dan alternatif yang ada.

Tabel IV. 10 Normalisasi Bobot

No	Kriteria (c)	Bobot (w_j)	Normalisasi ($w_j / \sum w_j$)
1	K1	60	0.6
2	K2	15	0.15
3	K3	10	0.1
4	K4	8	0.08
5	K5	7	0.07

(Sumber: Hasil Analisis, 2019)

Melakukan normalisasi dari bobot yang telah di tentukan

Tabel IV. 11 Nilai Kriteria Dari Setiap Alternatif

Kriteria / Alternatif	K1	K2	K3	K4	K5
A1	2	1	2	1	2
A2	1	2	1	2	1
A3	2	2	2	2	2
A4	1	2	2	2	2

(Sumber: Hasil Analisis, 2019)

Mengubah parameter yang keluar menjadi nilai parameter yang telah ditentukan pada tabel sebelumnya

Tabel IV. 12 Hasil Perhitungan Dengan Metode SMART

No	Alternatif	Kriteria	Nilai Utiliti	Bobot Normalisasi	Skor	Pernyataan
1	A1	K1	1	0.6	0.77	Butuh Penangan Khusus
		K2	0	0.15		
		K3	1	0.1		
		K4	0	0.08		
		K5	1	0.07		
2	A2	K1	0	0.6	0.23	Tidak Butuh Penangan Khusus
		K2	1	0.15		
		K3	0	0.1		
		K4	1	0.08		
		K5	0	0.07		
3	A3	K1	1	0.6	1	Butuh Penangan Khusus
		K2	1	0.15		
		K3	1	0.1		
		K4	1	0.08		
		K5	1	0.07		
4	A4	K1	0	0.6	0.4	Tidak Butuh Penangan Khusus
		K2	1	0.15		
		K3	1	0.1		
		K4	1	0.08		
		K5	1	0.07		

(Sumber: Hasil Analisis, 2019)

Setelah mengubah parameter kedalam nilai parameter maka selanjutnya dilakukan perhitungan nilai utility sekaligus perhitungan skor dengan bobot yang telah di normalisasi sebelumnya.

Tabel IV. 13 Hasil Keputusan

Alternatif	Keputusan
A1	Butuh Penangan Khusus
A2	Tidak Butuh Penangan Khusus
A3	Butuh Penangan Khusus
A4	Tidak Butuh Penangan Khusus

(Sumber: Hasil Analisis, 2019)

Diatas ini merupakan hasil keputusan dari perhitungan yang telah dilakukan dengan metode SMART.

BAB V

ANALISIS DAN PEMBAHASAN

5.1 Analisis Kebutuhan Sistem

Adapun kebutuhan-kebutuhan dari sistem pendukung keputusan tingkat kerusakan dies yang diusulkan dapat dilihat pada Tabel V.1 berikut:

Tabel V.1 Analisis Kebutuhan Sistem

No	Masalah	Kebutuhan User	Solusi	<i>Functional Requirement</i>
1	Terdapatnya kendala dalam pengambilan keputusan pada proses perbaikan <i>dies</i> , dikarenakan <i>staff</i> dari divisi <i>maintenance dies</i> kurang mengetahui proses perbaikan apa saja yang akan dilakukan apabila terjadi kerusakan.	Sistem yang dapat membantu pengambilan keputusan dalam proses perbaikan <i>dies</i> .	Membuat sistem pendukung keputusan yang dapat membantu dalam proses pengambilan keputusan proses perbaikan <i>dies</i> .	Sistem dapat memberikan saran untuk mendukung pengambilan keputusan dalam proses perbaikan <i>dies</i> .
2	Pencatatan hasil kegiatan perbaikan <i>dies</i> masih dilakukan pada buku besar sebagai media pencatatan, sehingga dapat mengakibatkan terjadinya kehilangan data dikarenakan rusaknya buku besar,	Sistem yang terintegrasi dengan basis data sebagai media penyimpanan sekaligus pengelolaan data terkait proses perbaikan <i>dies</i> .	Membuat sistem yang terintegrasi dengan basis data untuk menyimpan sekaligus mengelola data terkait proses perbaikan <i>dies</i> .	Sistem dapat mengelola dan menyimpan data perbaikan <i>dies</i> .

No	Masalah	Kebutuhan User	Solusi	<i>Functional Requirement</i>
	baik itu disebabkan oleh robeknya buku besar ataupun terkenanya oli dari tangan <i>staff</i> divisi <i>maintenance dies</i> pada saat melakukan pencatatan.			
3	Buku besar tempat pencatatan hasil kegiatan perbaikan <i>dies</i> diberikan kepada <i>engineering manager</i> sebagai laporan hasil perbaikan <i>dies</i> , yang mana mengakibatkan terhambatnya proses perekapan data dikarenakan tulisan yang kadang sulit dibaca oleh <i>engineering manager</i> .	Sistem yang dapat menghasilkan laporan hasil kegiatan perbaikan dies.	Membuat sistem yang mampu menghasilkan laporan hasil kegiatan perbaikan dies.	Sistem dapat menghasilkan laporan hasil kegiatan perbaikan dies.

(Sumber: Hasil Analisis, 2019)

5.5.1 *Non Functional Requirement*

Non functional requirement secara umum berisi batasan-batasan pada pelayan atau fungsi yang disediakan oleh sistem. Berikut adalah non functional requirement dari sistem pendukung keputusan proses perbaikan dies yang diusulkan.

1. Operasional.

Sistem dapat diakses oleh pengguna melalui *browser* yang mendukung format PHP seperti Google Chrome serta membutuhkan *web server* dan database seperti Apache dan MySQL.

2. Antarmuka (*Interface*).

Sistem cukup sederhana sehingga pengguna dapat dengan mudah memahami dan menggunakannya.

3. *Supportability*.

- Minimal RAM 2 GB.
- Terdapat sisa penyimpanan dalam *harddisk* minimal 64 GB.
- Minimal *processor* Intel Core 2.
- Menggunakan *mouse*, *keyboard* dan *monitor* sebagai media antarmuka.
- Menggunakan *printer* untuk mencetak laporan hasil kegiatan perbaikan *dies*.

5.2 Alur Sistem Pendukung Keputusan Proses Perbaikan *Dies*

Alur sistem pendukung keputusan proses perbaikan *dies* yang diusulkan antara lain sebagai berikut:

1. Divisi *production* akan membuat permintaan perbaikan *dies* yang nantinya akan dimasukkan kedalam *database*.
2. Divisi *maintenance dies* akan menerima permintaan perbaikan *dies* yang telah diinput oleh divisi *production*.
3. Setelah menerima permintaan perbaikan *dies*, divisi *maintenance dies* akan menganalisis tingkat proses perbaikan dies, apa saja solusi untuk melakukan proses perbaikan dies yang ada.
4. Jika dibutuhkan penanganan khusus maka divisi *maintenance dies* akan meminta persetujuan kepada *customer* selaku pemilik *dies* untuk melakukan perbaikan *dies* pada *workshop* rekanan, namun apabila tidak membutuhkan penanganan khusus maka divisi *maintenance dies* akan melakukan perbaikan *dies* pada perusahaan.

5. Apabila *customer* tidak menyetujui untuk dilakukan perbaikan pada *workshop* rekanan, maka *customer* akan membawa *dies* ke *workshop*nya sendiri yang nantinya akan dilakukan perbaikan pada *wokshop customer*.
6. Namun apabila *customer* menyetujui untuk dilakukan perbaikan pada *workshop* rekanan maka divisi *maintenance dies* akan membuat permintaan perbaikan *dies* pada rekanan.
7. Jika permintaan tidak disetujui maka divisi *maintenance dies* akan membuat permintaan perbaikan kembali pada rekanan lainnya, namun apabila permintaan disetujui maka divisi *maintenance dies* akan membawa *dies* ke *workshop* rekanan.
8. Setelah melakukan perbaikan, baik pada perusahaan, *workshop* rekanan maupun *workshop* customer divisi *maintenance dies* akan mengevaluasi kelayakan *dies* apakah *dies* sudah dalam kondisi yang layak atau belum.
9. Jika *dies* belum layak maka akan dilakukan perbaikan kembali hingga kondisi *dies* layak digunakan untuk proses produksi.
10. Namun apabila kondisi *dies* sudah layak digunakan untuk proses produksi divisi *maintenance dies* akan memasukan hasil kegiatan perbaikan *dies* yang nantinya akan dimasukan kedalam *database*.
11. Setelah memasukan hasil kegiatan perbaikan *dies*, *enggineering manager* akan menerima hasil kegiatan perbaikan *dies*.
12. *Engineering manager* dapat mencetak hasil kegiatan perbaikan *dies* sebagai bentuk laporan kegiatan perbaikan *dies*.

Gambaran mengenai proses perbaikan dies yang yang diusulkan dapat dilihat pada Gambar V.1 berikut:

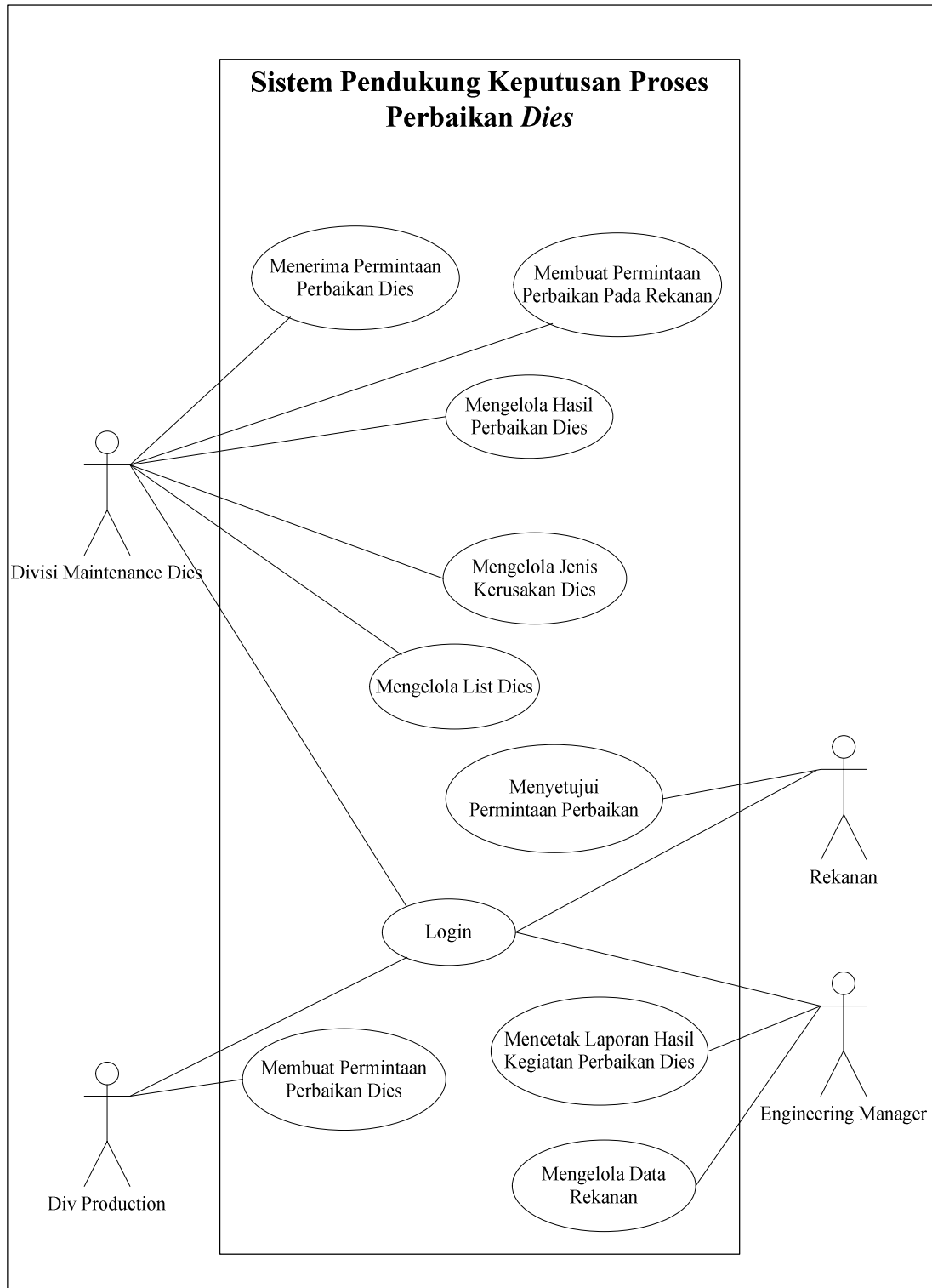
Gambar V.1 *Flowmap* Sistem Perndukung Keputusan Proses Perbaikan Dies Usulan
(Sumber: Hasil Analisis, 2019)

5.3 Pemodelan Sistem Pendukung Keputusan Proses Perbaikan Dies

Pada tahap ini dilakukan pemodelan sistem dengan menggunakan UML (*Unified Modelling Language*) di antaranya adalah *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, *Class Diagram*, dan *Deployment Diagram*.

5.3.1 Use Case Diagram

Use Case Diagram sistem pendukung keputusan tingkat kerusakan dies pada PT Nusa Indah Jaya Utama yang diusulkan dapat dilihat pada Gambar V.2.



Gambar V.2 Use Case Diagram Sistem Pendukung Keputusan Usulan
(Sumber: Hasil Analisis, 2019)

5.3.2 Use Case Description

Use Case Description dari sistem pendukung keputusan tingkat kerusakan *dies* yang diusulkan antara lain sebagai berikut:

1. Use Case Description Menerima Permintaan Perbaikan Dies

Tabel V.2 *Use Case Description* Menerima Permintaan Perbaikan Dies

Nama <i>Use Case</i>	Menerima Permintaan Perbaikan Dies
<i>Actor</i>	Divisi <i>Maintenance Dies</i>
<i>Normal Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu perbaikan dan sub-menu konfirmasi perbaikan. 2. Sistem akan menampilkan data permintaan perbaikan yang terdapat pada database. 3. Aktor akan dapat melihat daftar permintaan perbaikan yang ada. 4. Aktor dapat memilih tombol detail untuk melihat detail data dari permintaan perbaikan yang ada.

(Sumber: Hasil Analisis, 2019)

2. Use Case Description Membuat Permintaan Perbaikan Dies Pada Rekanan

Tabel V.3 *Use Case Description* Membuat Permintaan Perbaikan Dies Pada Rekanan

Nama <i>Use Case</i>	Membuat Permintaan Perbaikan Dies Pada Rekanan
<i>Actor</i>	Divisi <i>Maintenance Dies</i>
<i>Normal Flow</i>	<ol style="list-style-type: none"> 1. Actor memilih menu permintaan perbaikan. 2. Sistem akan menampilkan daftar permintaan yang sudah ada. 3. Actor dapat memilih tombol tambah data permintaan perbaikan dies. 4. Sistem akan menampilkan form permintaan perbaikan dies pada rekanan.

Tabel V.3 *Use Case Description* Membuat Permintaan Perbaikan Dies Pada Rekanan (lanjutan)

<i>Normal Flow</i>	<p>5. Actor akan memasukan data permintaan perbaikan serta data rekanan yang akan diberikan permintaan perbaikan dies yang selanjutnya dapat memilih tombol simpan untuk menyimpan data dan tombol cancel untuk membatalkan.</p> <p>6. Sistem akan memasukan data yang telah diisi kedalam database.</p>
--------------------	--

(Sumber: Hasil Analisis, 2019)

3. *Use Case Description* Mengelola Hasil Perbaikan Dies

Tabel V.4 *Use Case Description* Mengelola Hasil Perbaikan Dies

Nama <i>Use Case</i>	Mengelola Hasil Perbaikan Dies
<i>Actor</i>	Divisi <i>Maintenance Dies</i>
<i>Normal Flow</i>	<p>1. Actor memilih menu validasi permintaan perbaikan dies.</p> <p>2. Sistem akan menampilkan data permintaan perbaikan yang terdapat pada database.</p> <p>3. Actor akan dapat melihat daftar permintaan perbaikan yang ada.</p> <p>4. Actor dapat memilih tombol validasi untuk mengelola data hasil perbaikan dies.</p> <p>5. Sistem akan menampilkan form tanggal dies diperbaiki.</p> <p>6. Aktor akan memasukan tanggal dies diperbaiki.</p> <p>7. Aktor dapat memilih tombol validasi untuk memvalidasi dan tombol cancel untuk membatalkan.</p>

(Sumber: Hasil Analisis, 2019)

4. *Use Case Description* Mengelola List Dies

Tabel V.5 *Use Case Description* Mengelola List Dies

Nama <i>Use Case</i>	Mengelola List Dies
<i>Actor</i>	Divisi <i>Maintenance Dies</i>
<i>Normal Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu dies 2. Sistem akan menampilkan daftar dies yang sudah ada. 3. Aktor dapat memilih tombol tambah data dies untuk menambahkan data, ubah untuk mengubah data yang telah ada, serta hapus untuk menghapus data yang ada. 4. Jika aktor memilih tombol tambah data dies sistem akan menampilkan form untuk diisi 5. Setelah aktor mengisi form, aktor dapat memilih tombol simpan untuk menyimpan data serta tombol cancel untuk membatalkan

(Sumber: Hasil Analisis, 2019)

5. *Use Case Description* Mengelola Jenis Kerusakan Dies

Tabel V.6 *Use Case Description* Mengelola Jenis Kerusakan Dies

Nama <i>Use Case</i>	Mengelola Jenis Kerusakan Dies
<i>Actor</i>	Divisi <i>Maintenance Dies</i>
<i>Normal Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu kerusakan 2. Sistem akan menampilkan daftar kerusakan yang ada. 3. Aktor dapat memilih tombol tambah data kerusakan untuk menambahkan data, ubah untuk mengubah data yang telah ada, serta hapus untuk menghapus data yang ada. 4. Jika aktor memilih tombol tambah data kerusakan sistem akan menampilkan form untuk diisi

Tabel V.6 *Use Case Description* Mengelola Jenis Kerusakan Dies (lanjutan)

<i>Normal Flow</i>	<p>5. Setelah aktor mengisi form, aktor dapat memilih tombol simpan untuk menyimpan data serta tombol cancel untuk membatalkan</p> <p>6. Actor akan memasukan data permintaan perbaikan serta data rekanan yang akan diberikan permintaan perbaikan dies yang selanjutnya dapat memilih tombol simpan untuk menyimpan data dan tombol cancel untuk membatalkan.</p> <p>7. Sistem akan memasukan data yang telah diisi kedalam database.</p>
--------------------	---

(Sumber: Hasil Analisis, 2019)

6. *Use Case Description* LoginTabel V.7 *Use Case Description* Login

Nama <i>Use Case</i>	Login
<i>Actor</i>	Divisi <i>Maintenance Dies</i> , <i>Engineering Manager</i> , Divisi <i>Production</i>
<i>Normal Flow</i>	<p>1. <i>Actor</i> membuka aplikasi.</p> <p>2. Sistem akan menampilkan halaman <i>login</i>.</p> <p>3. <i>Actor</i> memasukkan <i>username</i> dan <i>password</i>.</p> <p>4. <i>Actor</i> mengklik tombol <i>Login</i>.</p> <p>5. Sistem akan memvalidasi <i>username</i> dan <i>password</i>.</p> <p>6. Jika <i>username</i> dan <i>password</i> valid, sistem akan menampilkan halaman utama.</p> <p>7. Jika <i>username</i> dan <i>password</i> tidak valid, sistem akan menampilkan pesan kesalahan.</p>

(Sumber: Hasil Analisis, 2019)

7. *Use Case Description* Mengelola Data Rekanan

Tabel V.8 *Use Case Description* Mengelola Data Rekanan

Nama <i>Use Case</i>	Mengelola Data Rekanan
<i>Actor</i>	<i>Engineering Manager</i>
<i>Normal Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu rekanan 2. Sistem akan menampilkan daftar rekanan yang ada. 3. Aktor dapat memilih tombol tambah data rekanan untuk menambahkan data, ubah untuk mengubah data yang telah ada, serta hapus untuk menghapus data yang ada. 4. Jika aktor memilih tombol tambah data rekanan sistem akan menampilkan form untuk diisi 5. Setelah aktor mengisi form, aktor dapat memilih tombol simpan untuk menyimpan data serta tombol cancel untuk membatalkan

(Sumber: Hasil Analisis, 2019)

8. *Use Case Description* Menerima Laporan Hasil Perbaikan Dies

Tabel V.9 *Use Case Description* Menerima Laporan Hasil Perbaikan Dies

Nama <i>Use Case</i>	Menerima Laporan Hasil Perbaikan Dies
<i>Actor</i>	<i>Engineering Manager</i>
<i>Normal Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu perbaikan di setujui. 2. Sistem akan menampilkan daftar perbaikan yang telah dilakukan. 3. Aktor dapat memilih menu detail untuk melihat detail data. 4. Sistem akan menampilkan detail data dari daftar perbaikan yg dipilih.

(Sumber: Hasil Analisis, 2019)

9. *Use Case Description* Membuat Permintaan Perbaikan Dies

Tabel V.10 *Use Case Description* Membuat Permintaan Perbaikan Dies

Nama <i>Use Case</i>	Membuat Permintaan Perbaikan Dies
<i>Actor</i>	Divisi Production
<i>Normal Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu permintaan perbaikan 2. Sistem akan menampilkan daftar permintaan yang telah ada. 3. Aktor dapat memilih menu tambah data untuk menambah data permintaan. 4. Sistem akan menampilkan form isian data. 5. Aktor memilih menu simpan untuk menyimpan data

(Sumber: Hasil Analisis, 2019)

10. *Use Case Description* Menyetujui Permintaan Perbaikan

Tabel V.11 *Use Case Description* Menyetujui Permintaan Perbaikan

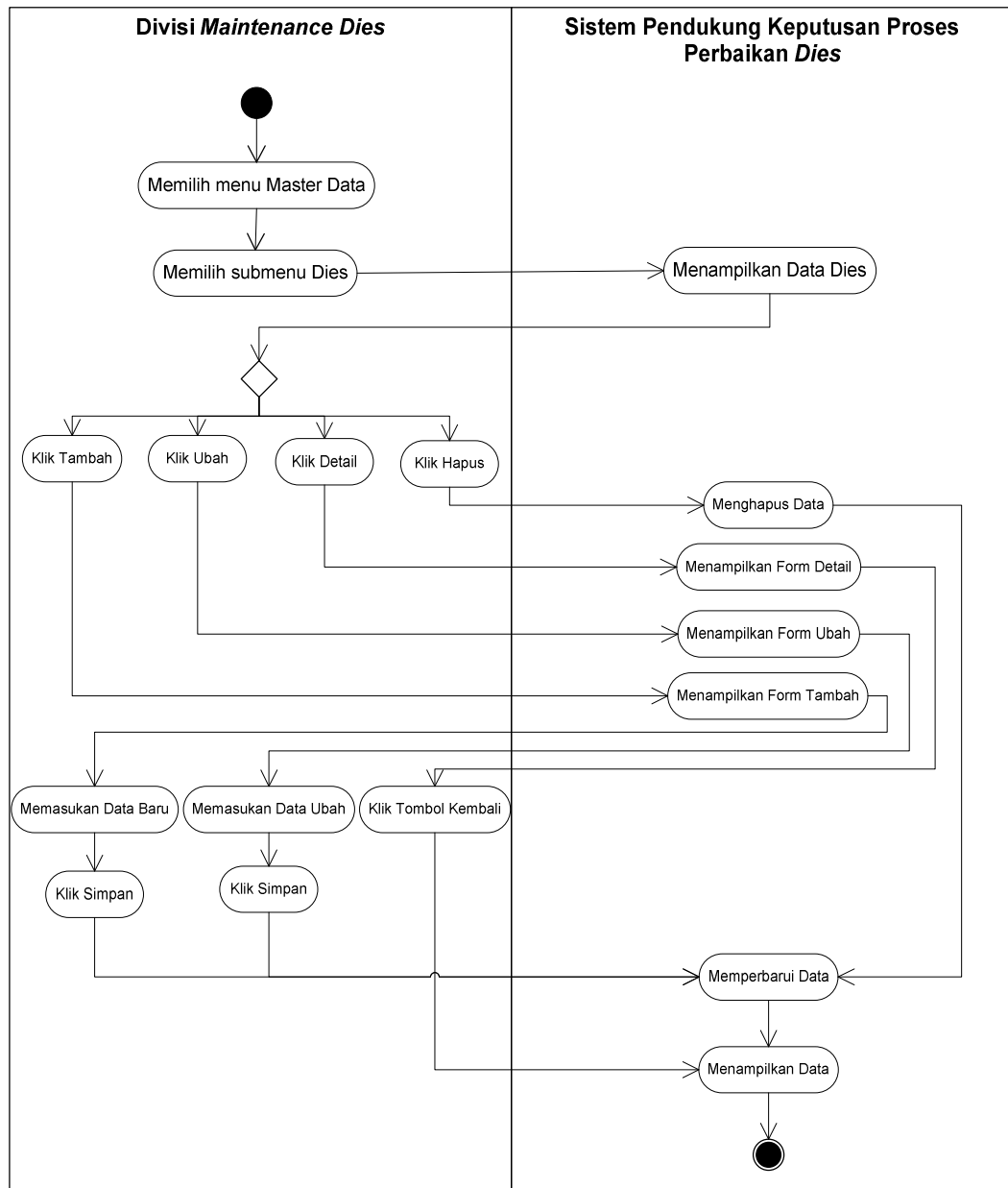
Nama <i>Use Case</i>	Menyetujui Permintaan Perbaikan
<i>Actor</i>	Rekanan
<i>Normal Flow</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu permintaan perbaikan 2. Sistem akan menampilkan permintaan yang telah dikirimkan oleh divisi <i>maintenance dies</i>. 3. Aktor dapat memilih menu terima, tolak dan detail. 4. Sistem akan mengubah status apabila aktor memilih tombol terima atau tolak.

(Sumber: Hasil Analisis, 2019)

5.4 *Activity Diagram*

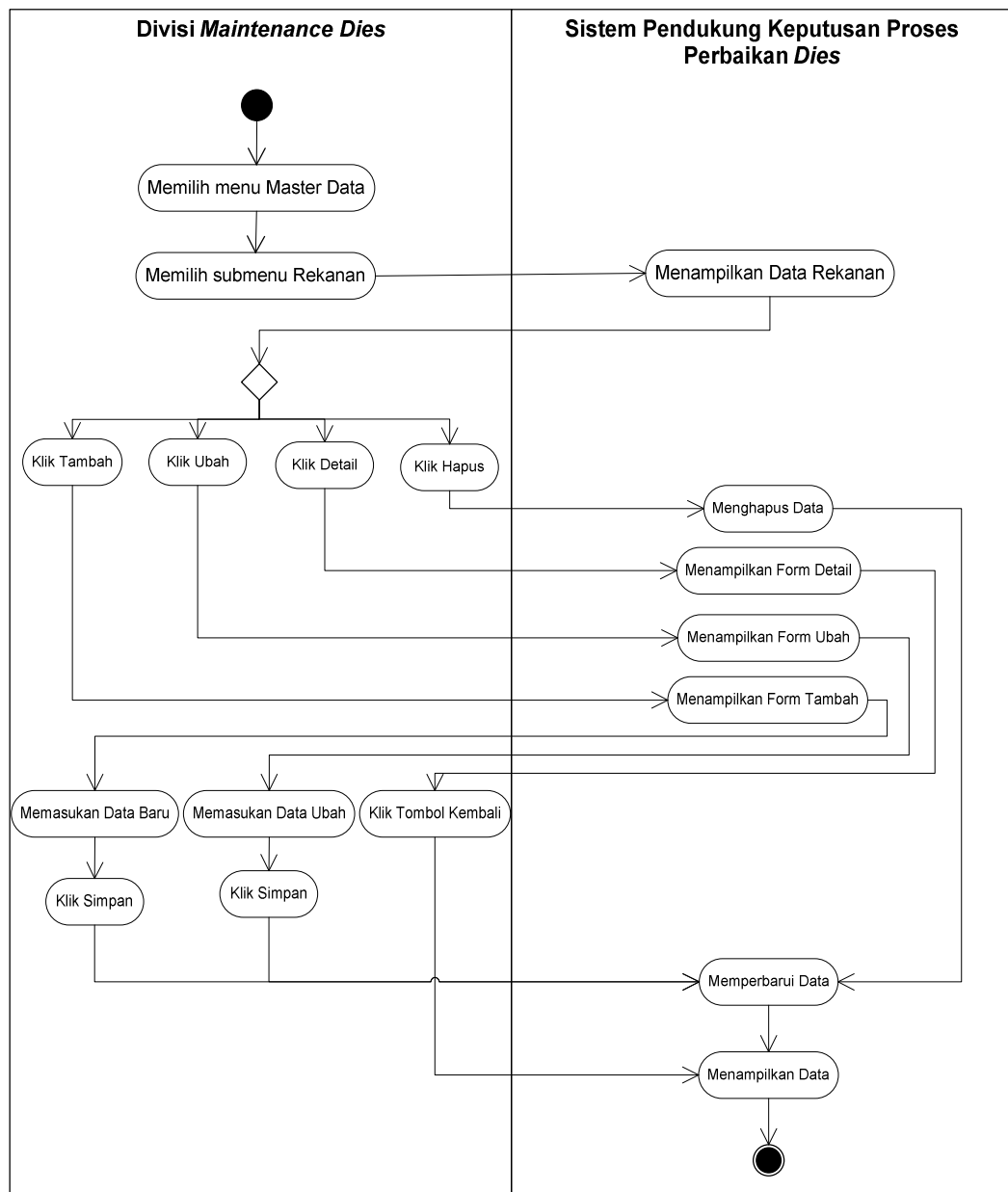
Activity Diagram menggambarkan urutan aktivitas proses bisnis pada suatu sistem. *Activity Diagram* sistem pendukung keputusan proses perbaikan *dies* pada PT Nusa Indah Jaya Utama yang diusulkan antara lain sebagai berikut:

1. Activity Diagram Mengelola Data Master Dies



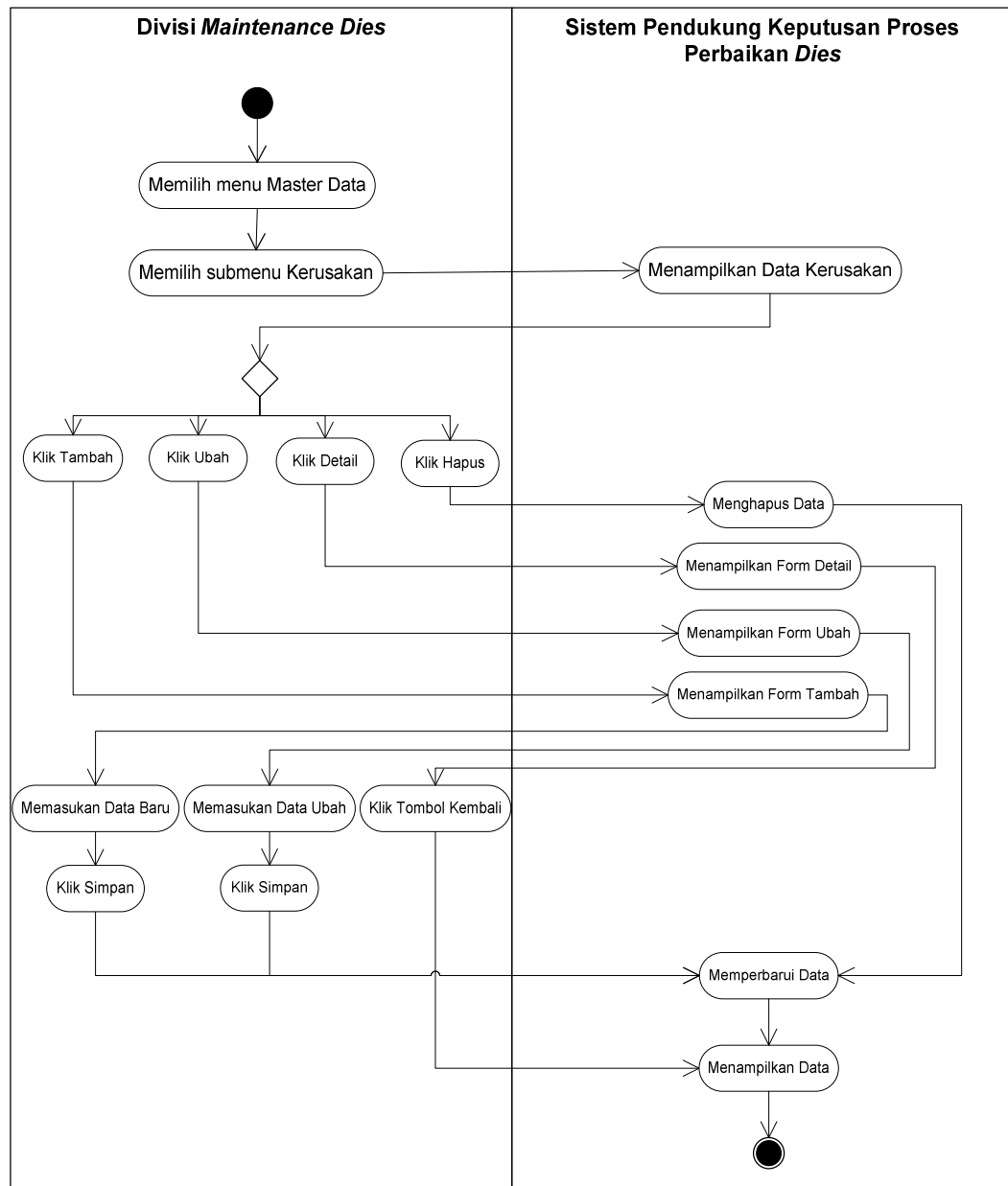
Gambar V.3 Activity Diagram Mengelola Data Master Dies
(Sumber: Hasil Analisis, 2019)

2. Activity Diagram Mengelola Data Master Rekanan



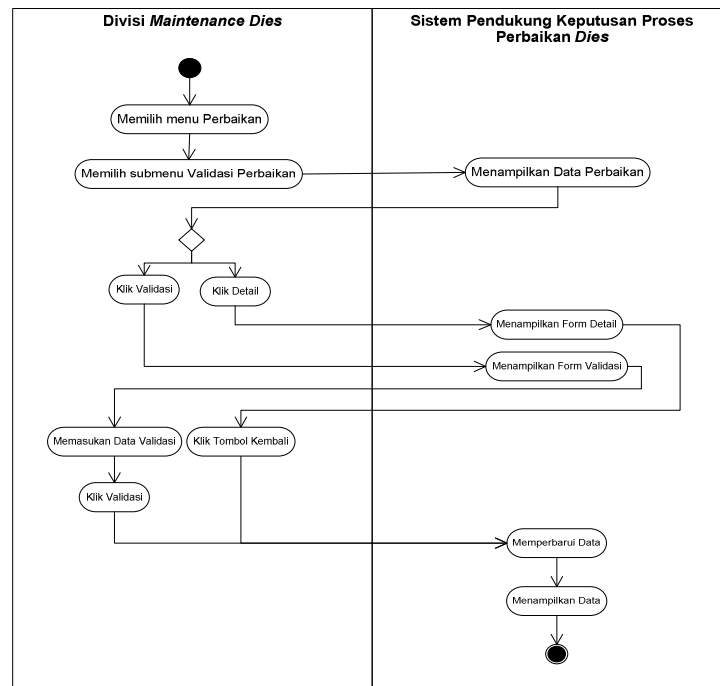
Gambar V.4 Activity Diagram Mengelola Data Master Rekanan
(Sumber: Hasil Analisis, 2019)

3. Activity Diagram Mengelola Data Master Kerusakan



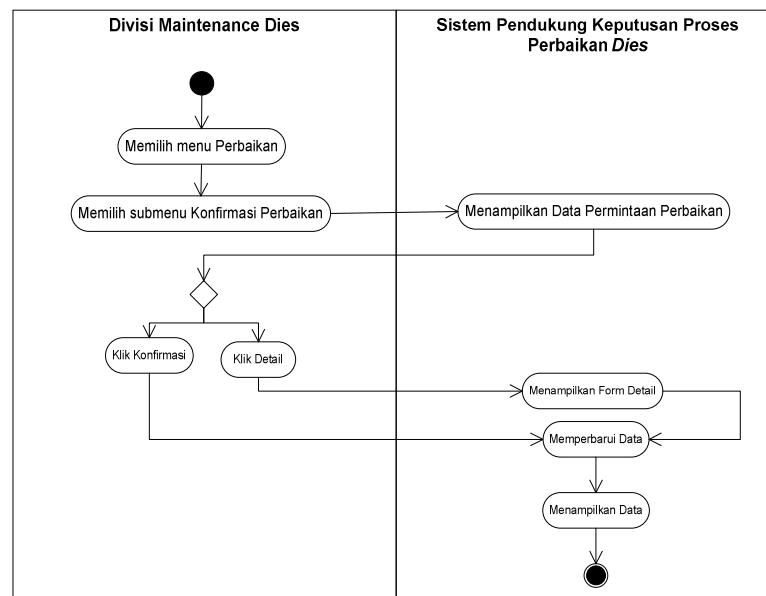
Gambar V.5 Activity Diagram Mengelola Data Master Kerusakan
(Sumber: Hasil Analisis, 2019)

4. Activity Diagram Mengelola Hasil Perbaikan



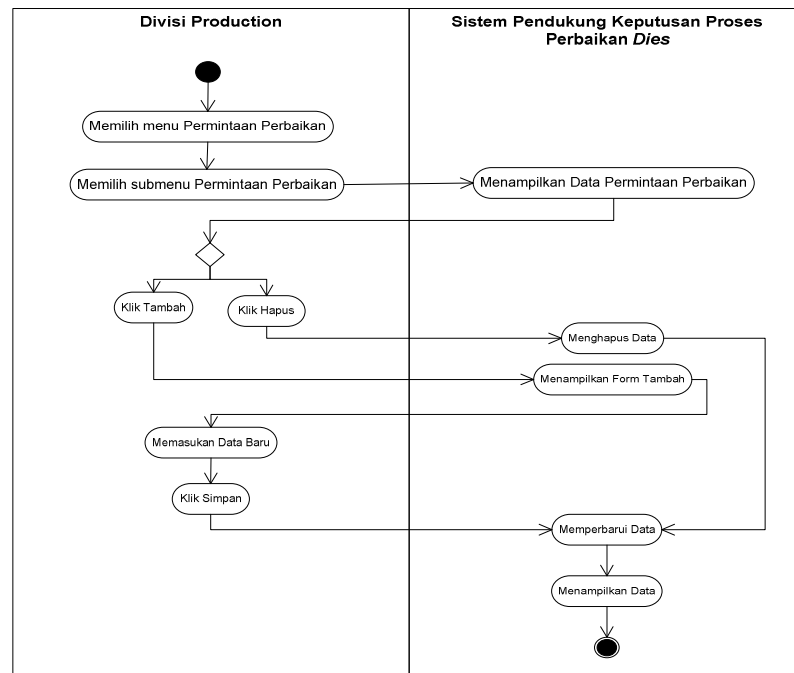
Gambar V.6 Activity Diagram Mengelola Hasil Perbaikan
(Sumber: Hasil Analisis, 2019)

5. Activity Diagram Menerima Permintaan Perbaikan



Gambar V.7 Activity Diagram Menerima Pemintaan Perbaikan
(Sumber: Hasil Analisis, 2019)

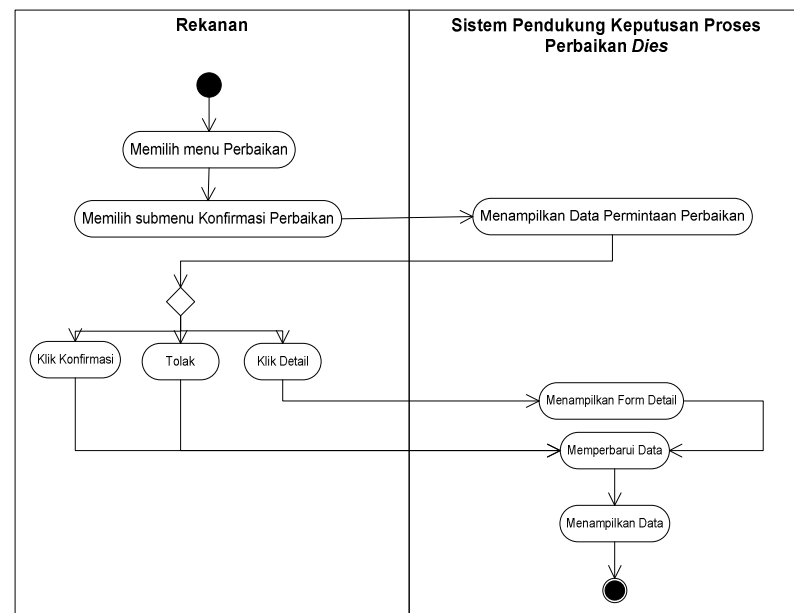
6. Activity Diagram Permintaan Perbaikan



Gambar V.8 Activity Diagram Permintaan Perbaikan

(Sumber: Hasil Analisis, 2019)

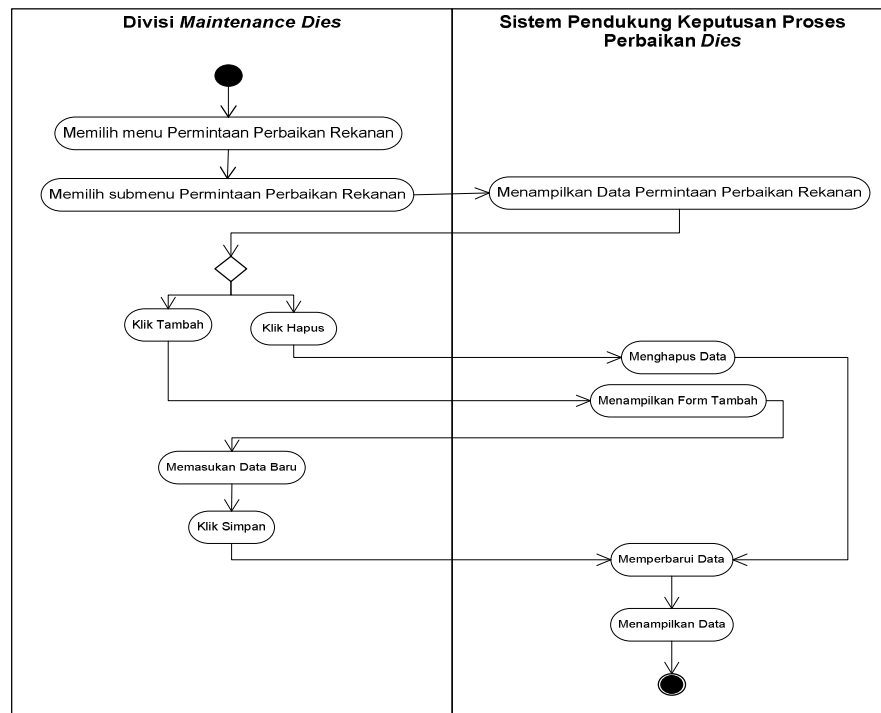
7. Activity Diagram Menyetujui Permintaan Perbaikan Rekanan



Gambar V.9 Activity Diagram Menyetujui Permintaan Perbaikan Rekanan

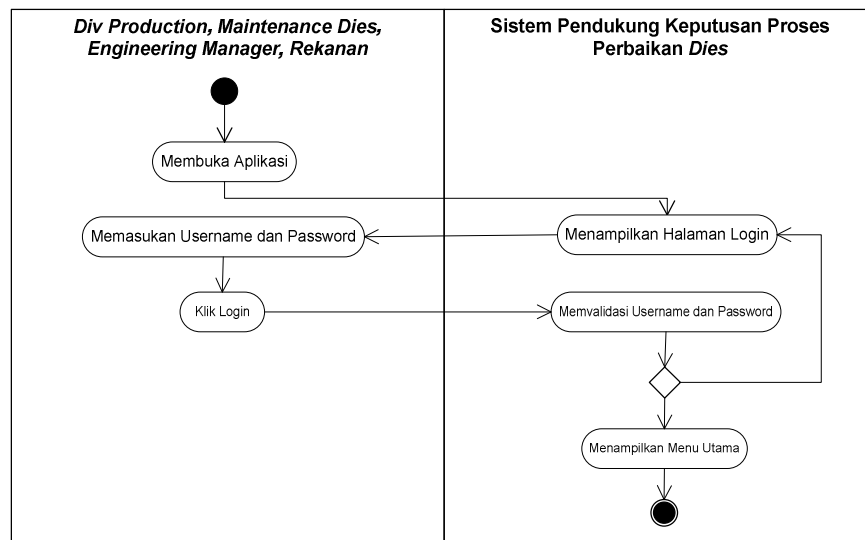
(Sumber: Hasil Analisis, 2019)

8. Activity Diagram Permintaan Perbaikan Rekanan



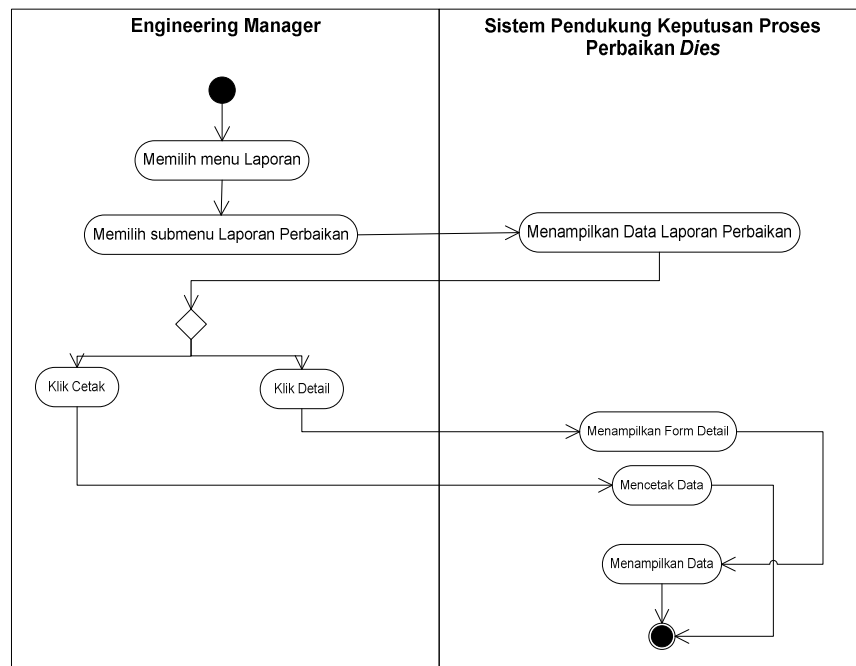
Gambar V.10 Activity Diagram Permintaan Perbaikan Rekanan
(Sumber: Hasil Analisis, 2019)

9. Activity Diagram Melakukan Login



Gambar V.11 Activity Diagram Melakukan Login
(Sumber: Hasil Analisis, 2019)

10. Activity Diagram Mencetak Data



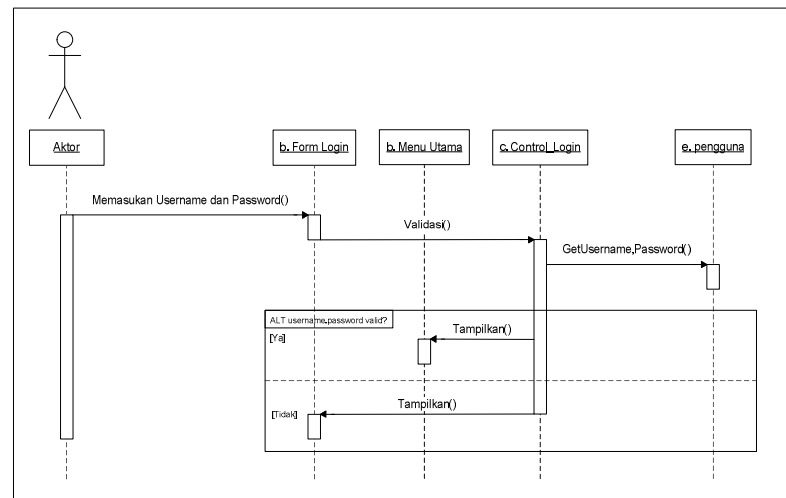
Gambar V.12 Activity Diagram Mencetak Data
(Sumber: Hasil Analisis, 2019)

5.5 Sequence Diagram

Sequence Diagram menggambarkan interaksi yang terjadi antar objek pada suatu *Use Case Diagram*. *Sequence Diagram* sistem pendukung keputusan proses perbaikan *dies* yang diusulkan antara lain sebagai berikut:

1. Sequence Diagram Login

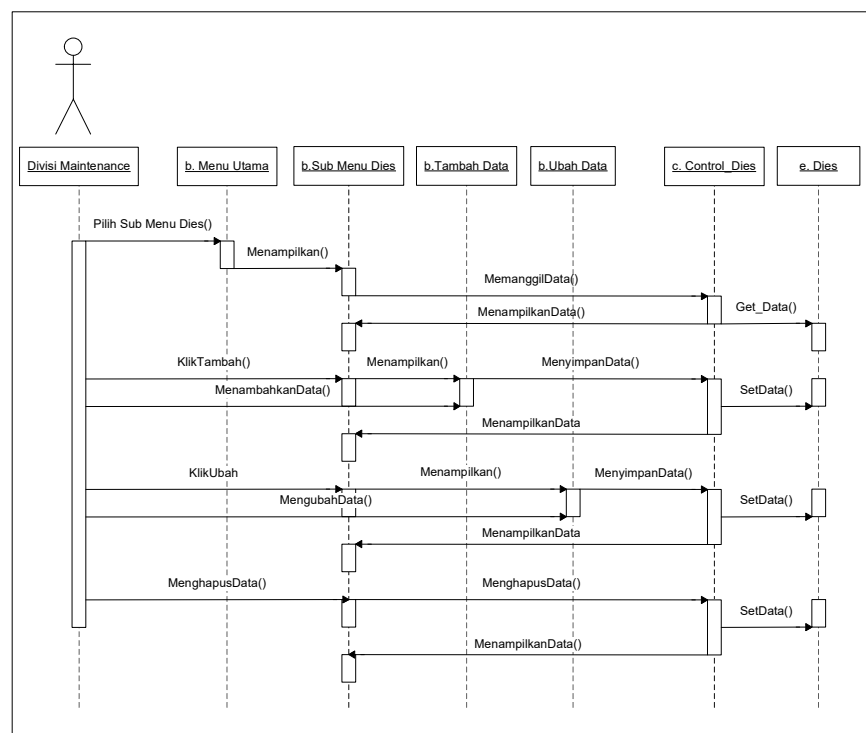
Sequence Diagram Login merupakan sebuah diagram *sequence* yang berfungsi untuk menjelaskan interaksi objek-objek dalam proses *login*. Di dalam sistem informasi penjadwalan produksi ini, *login* diperlukan agar seorang pengguna dapat mengakses sistem tersebut. Berikut merupakan *sequence diagram* dari *use case login*:

Gambar V.13 *Sequence Diagram Login*

Sumber: Hasil Analisis(2019)

2. *Sequence Diagram Mengelola Data Dies*

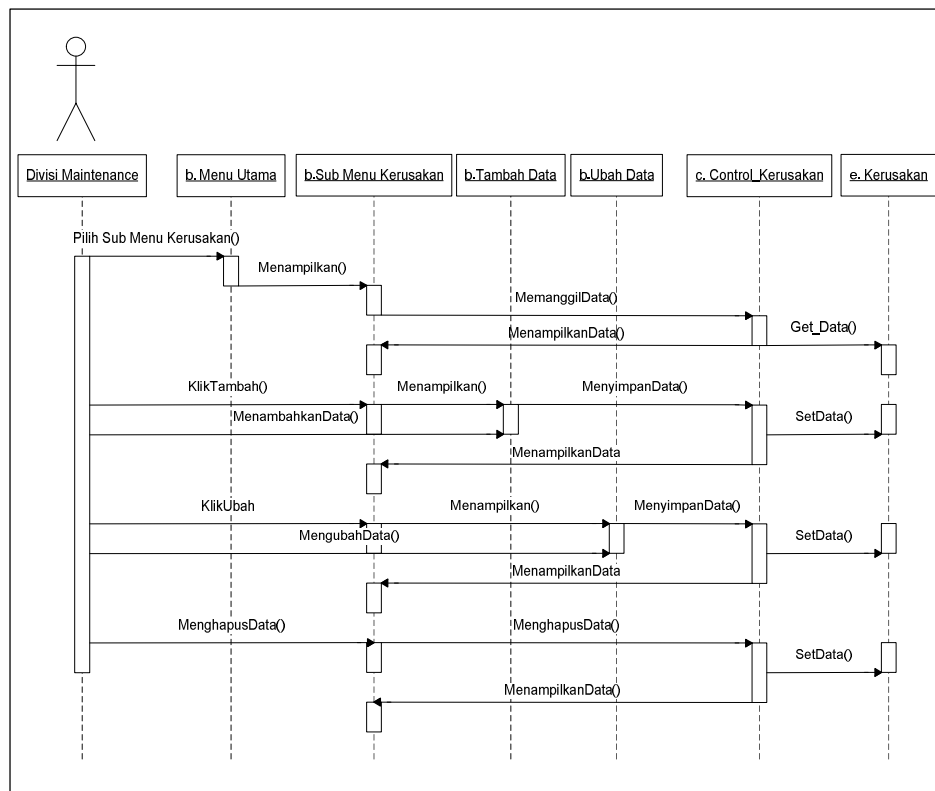
Sequence Diagram Mengelola Data Dies merupakan sebuah diagram *sequence* yang berfungsi untuk menjelaskan interaksi objek-objek dalam proses mengelola data *dies*. Berikut merupakan *sequence diagram* dari *use case* mengelola data *dies*:

Gambar V.14 *Sequence Diagram Login*

Sumber: Hasil Analisis(2019)

3. *Sequence Diagram* Mengelola Data Kerusakan

Sequence Diagram Mengelola Data Kerusakan merupakan sebuah diagram *sequence* yang berfungsi untuk menjelaskan interaksi objek-objek dalam proses mengelola data kerusakan. berikut merupakan *sequence diagram* dari *use case* mengelola data kerusakan:

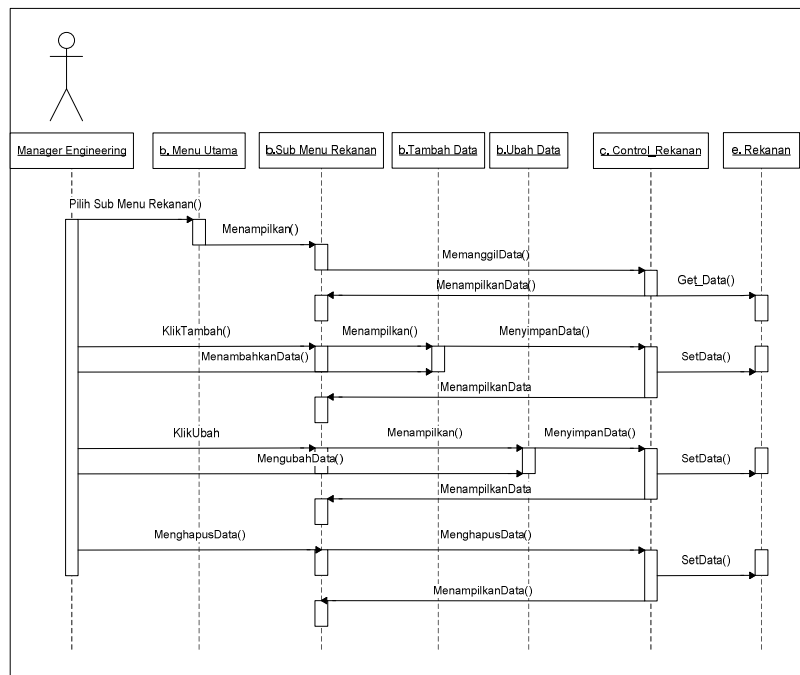


Gambar V.15 *Sequence Diagram* Mengelola Data Kerusakan

Sumber: Hasil Analisis(2019)

4. *Sequence Diagram* Mengelola Data Rekanan

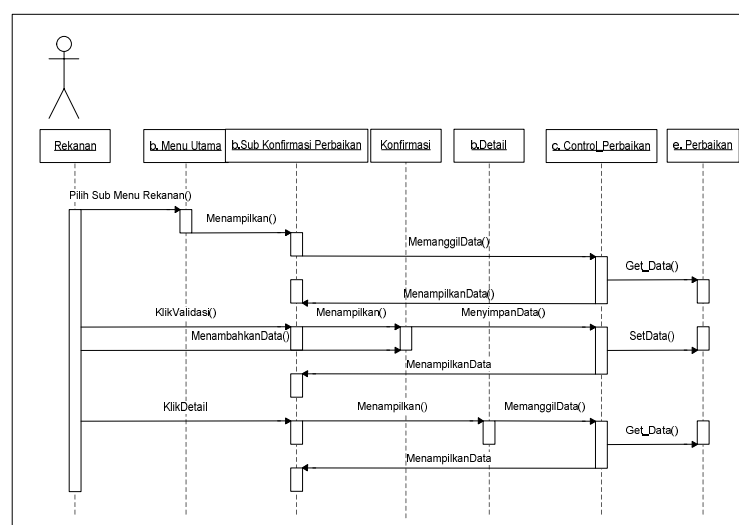
Sequence Diagram Mengelola Data Rekanan merupakan sebuah diagram *sequence* yang berfungsi untuk menjelaskan interaksi objek-objek dalam proses mengelola data rekanan. berikut merupakan *sequence diagram* dari *use case* mengelola data rekanan:

Gambar V.16 *Sequence Diagram* Mengelola Data Rekanan

Sumber: Hasil Analisis(2019)

5. *Sequence Diagram* Menerima Perbaikan Rekanan

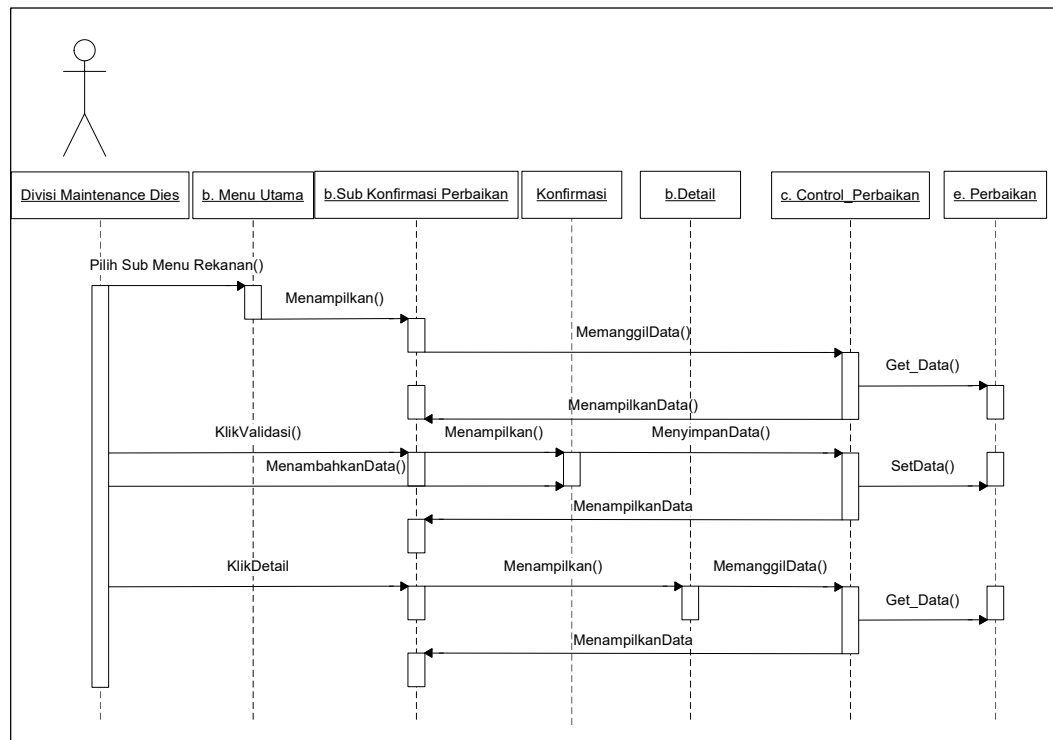
Sequence Diagram Menerima Perbaikan Rekanan merupakan sebuah diagram *sequence* yang berfungsi untuk menjelaskan interaksi objek-objek dalam proses menerima perbaikan rekanan. berikut merupakan *sequence diagram* dari *use case* menerima perbaikan rekanan:

Gambar V.17 *Sequence Diagram* Menerima Perbaikan Rekanan

Sumber: Hasil Analisis(2019)

6. *Sequence Diagram Menerima Permintaan Perbaikan*

Sequence Diagram Menerima Permintaan Perbaikan merupakan sebuah diagram *sequence* yang berfungsi untuk menjelaskan interaksi objek-objek dalam proses menerima permintaan perbaikan. berikut merupakan *sequence diagram* dari *use case* menerima permintaan perbaikan:

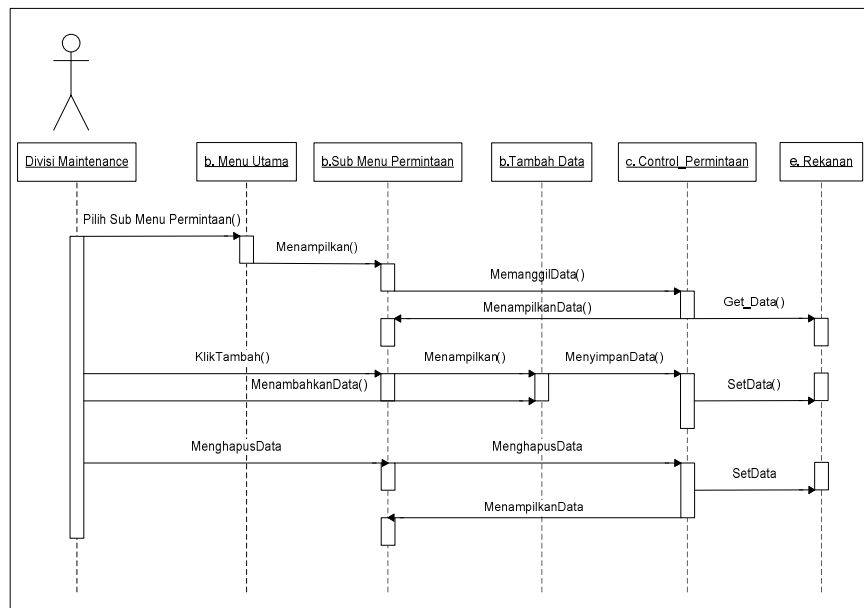


Gambar V.18 *Sequence Diagram Menerima Permintaan Perbaikan*

Sumber: Hasil Analisis(2019)

7. *Sequence Diagram Permintaan Perbaikan Rekanan*

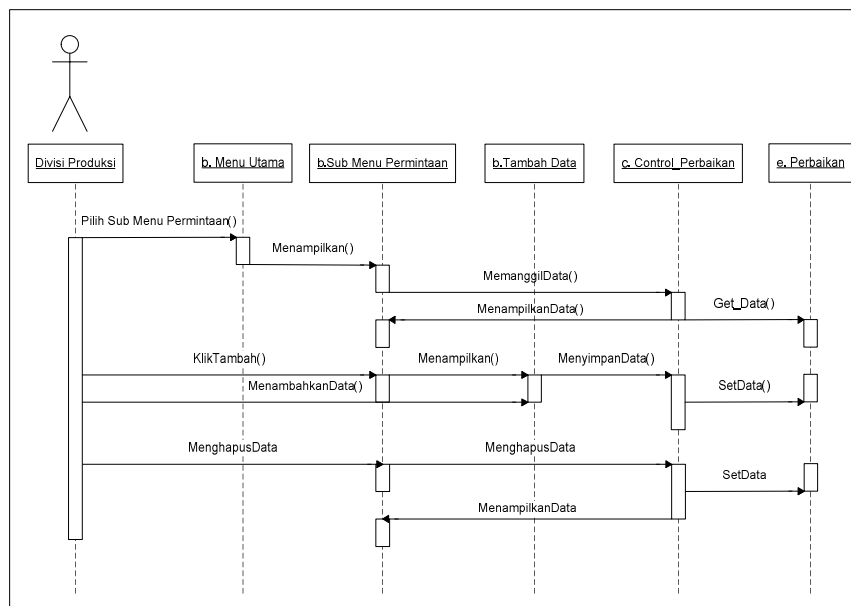
Sequence Diagram Permintaan Perbaikan Rekanan merupakan sebuah diagram *sequence* yang berfungsi untuk menjelaskan interaksi objek-objek dalam proses permintaan perbaikan rekanan berikut merupakan *sequence diagram* dari *use case* permintaan perbaikan rekanan:

Gambar V.19 *Sequence Diagram* Permintaan Perbaikan Rekanan

Sumber: Hasil Analisis(2019)

8. *Sequence Diagram* Permintaan Perbaikan

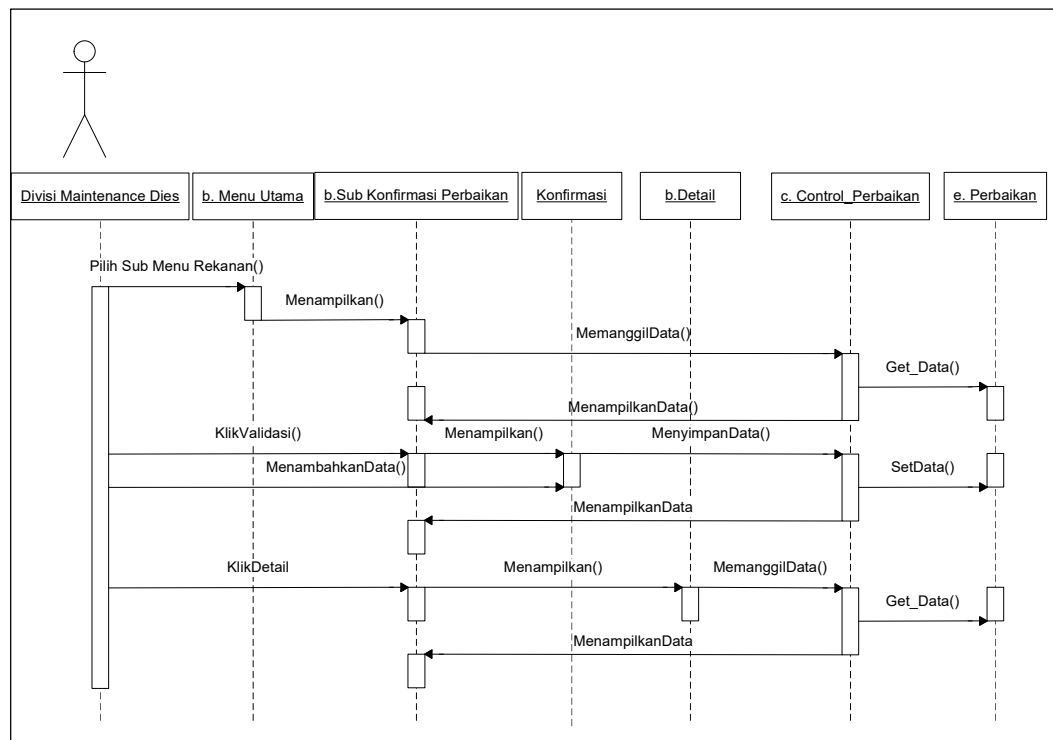
Sequence Diagram Permintaan Perbaikan merupakan sebuah diagram *sequence* yang berfungsi untuk menjelaskan interaksi objek-objek dalam proses permintaan perbaikan rekanan. berikut merupakan *sequence diagram* dari *use case* permintaan perbaikan:

Gambar V.20 *Sequence Diagram* Permintaan Perbaikan

Sumber: Hasil Analisis(2019)

9. *Sequence Diagram* Mengelola Hasil Perbaikan

Sequence Diagram Mengelola Hasil Perbaikan merupakan sebuah diagram *sequence* yang berfungsi untuk menjelaskan interaksi objek-objek dalam proses mengelola hasil perbaikan. berikut merupakan *sequence diagram* dari *use case* mengelola hasil perbaikan:

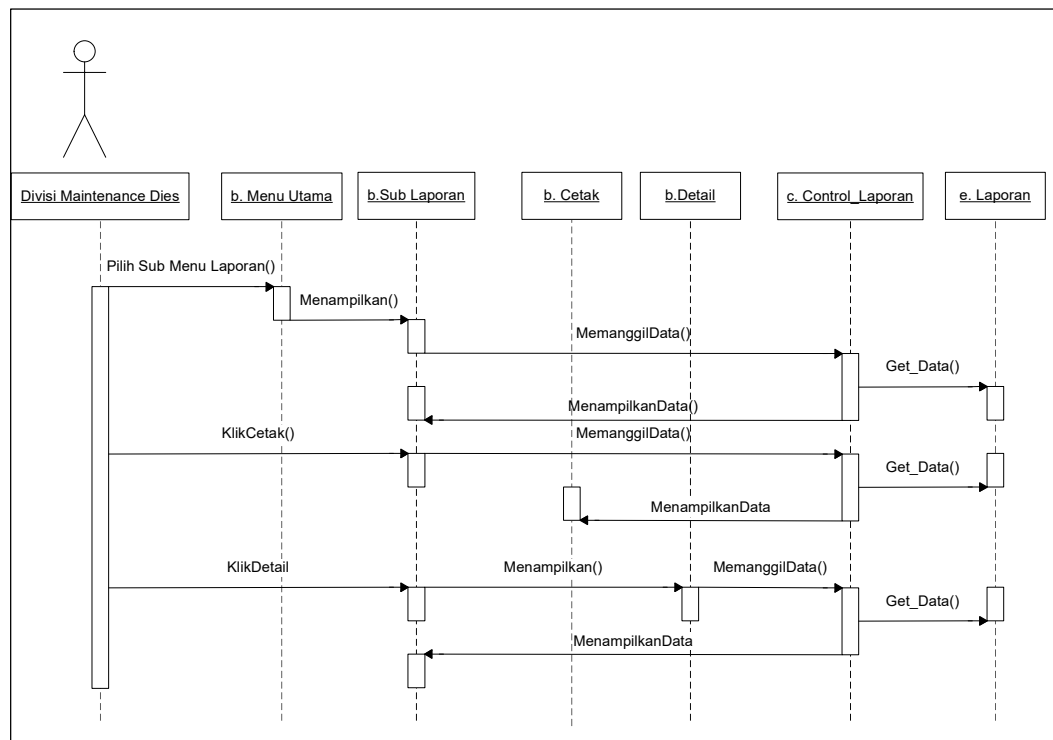


Gambar V.21 *Sequence Diagram* Mengelola Hasil Perbaikan

Sumber: Hasil Analisis(2019)

10. *Sequence Diagram* Cetak Laporan

Sequence Diagram Cetak Laporan merupakan sebuah diagram *sequence* yang berfungsi untuk menjelaskan interaksi objek-objek dalam proses cetak laporan. berikut merupakan *sequence diagram* dari *use case* cetak laporan:

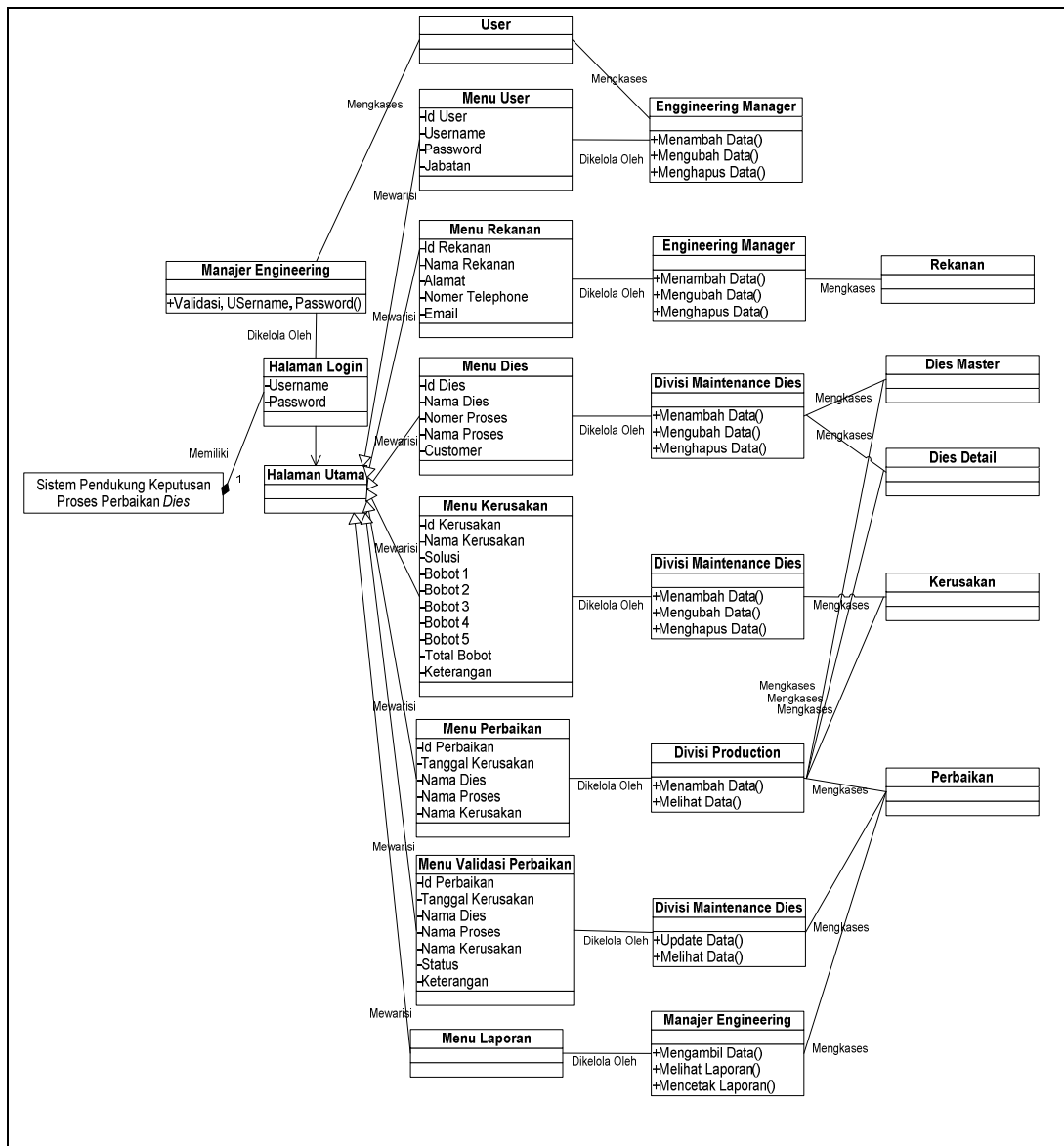


Gambar V.22 *Sequence Diagram* Cetak Laporan

Sumber: Hasil Analisis(2019)

5.6 *Class Diagram*

Class Diagram menggambarkan hubungan antar kelas di dalam sistem. *Class Diagram* sistem pendukung keputusan proses perbaikan *dies* yang diusulkan dapat dilihat pada Gambar V.23.

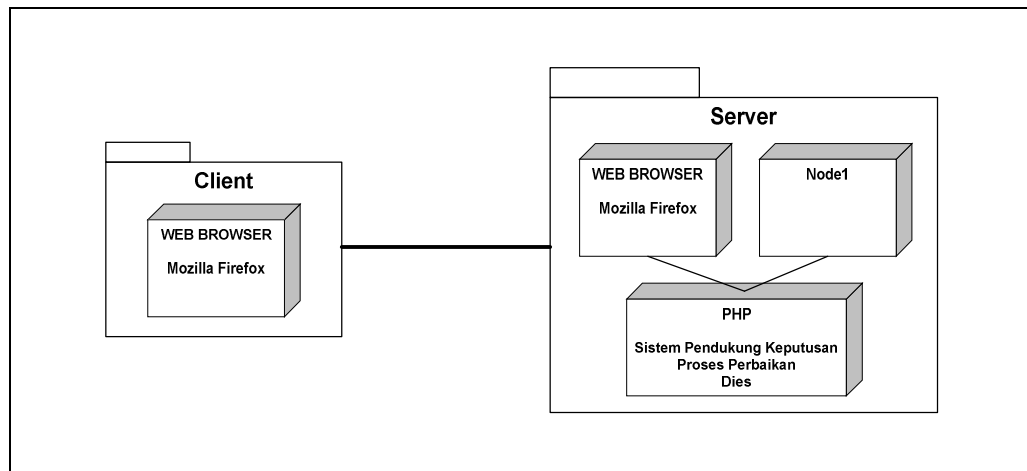


Gambar V.23 Class Diagram Sistem Informasi Usulan

(Sumber: Hasil Analisis, 2019)

5.7 Deployment Diagram

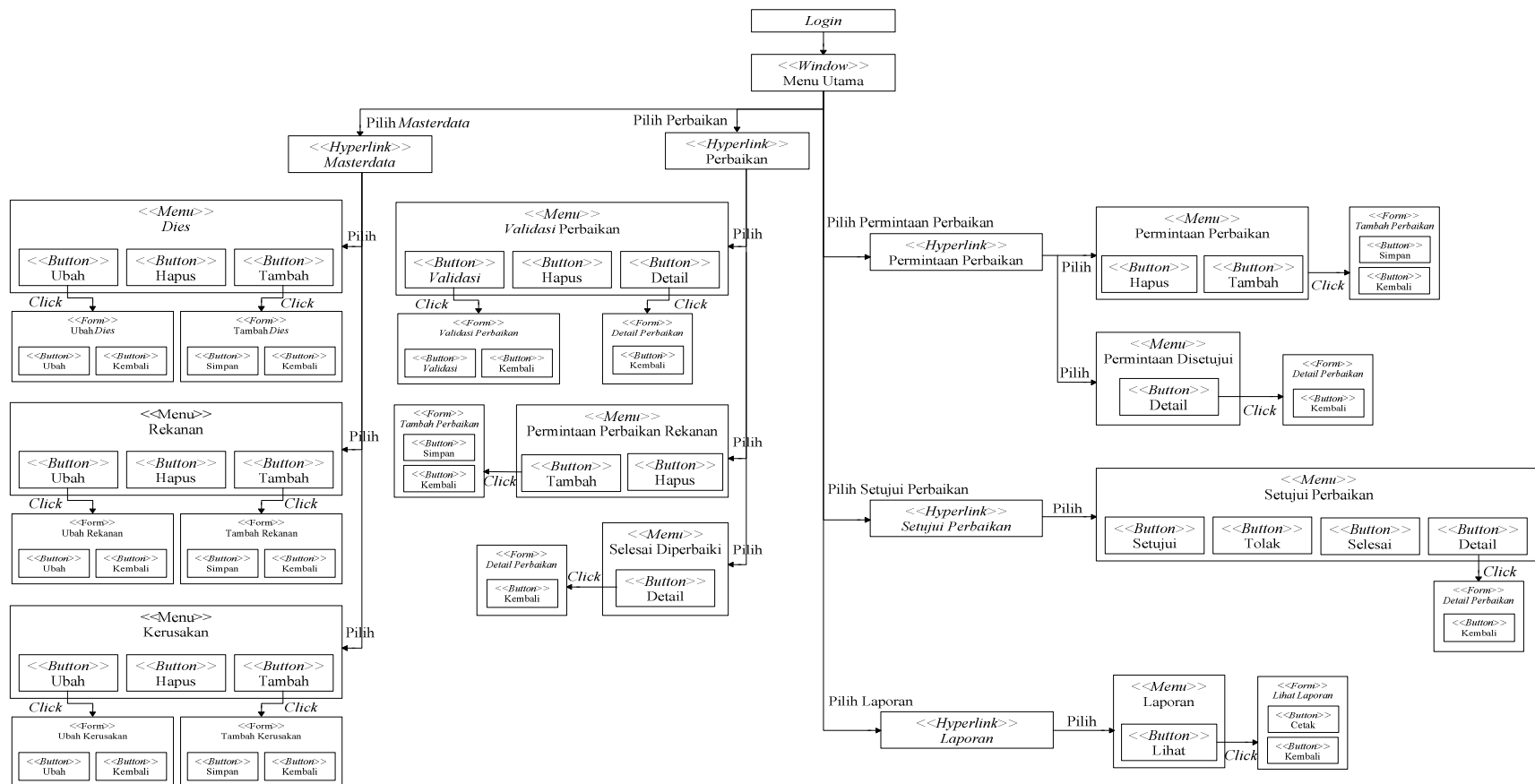
Deployment Diagram menunjukkan komponen perangkat lunak yang digunakan di dalam arsitektur fisik. *Deployment Diagram* sistem pendukung keputusan proses perbaikan *dies* usulan dapat dilihat pada Gambar V.24.



Gambar V.24 *Deployment Diagram* Sistem Pendukung Keputusan Usulan
(Sumber: Hasil Analisis, 2019)

5.8 *Windows Navigation Diagram (WND)*

Windows Navigation Diagram (WND) digunakan untuk mendokumentasikan sebuah struktur yang menggambarkan hubungan antar fungsi dalam program secara hirarki. WND sistem pendukung keputusan proses perbaikan *dies* yang diusulkan dapat dilihat pada Gambar V.25 berikut:



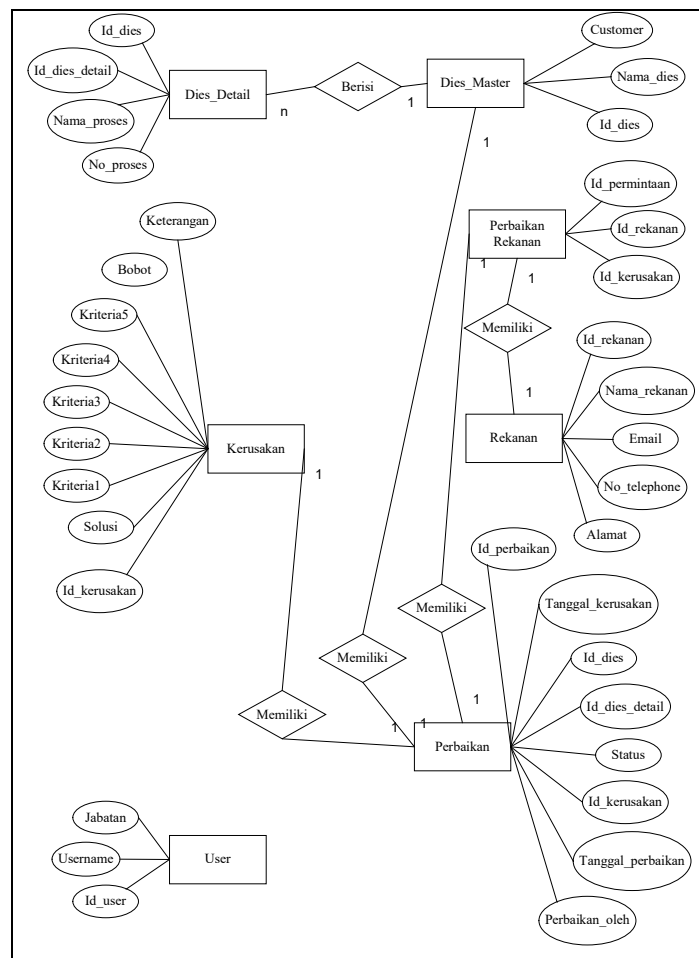
Gambar V.25 WND Sistem Usulan
(Sumber: Hasil Analisis, 2019)

5.9 Pemodelan Data Sistem Pendukung Keputusan Usulan

Pemodelan data merupakan metode yang digunakan untuk menentukan dan menganalisis persyaratan data yang diperlukan untuk mendukung proses bisnis suatu organisasi. Pemodelan data sistem pendukung keputusan proses perbaikan *dies* yang diusulkan dibuat dengan menggunakan *Entity Relationship Diagram* (ERD) dan kamus data.

5.9.1 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) merupakan salah satu model yang digunakan untuk mendesain *database* dengan tujuan menggambarkan data yang berelasi pada sebuah *database*. ERD sistem pendukung keputusan proses perbaikan *dies* yang diusulkan dapat dilihat pada Gambar V.26.



Gambar V.26 ERD Sistem Informasi Usulan
(Sumber: Hasil Analisis, 2019)

5.9.2 Kamus Data

Kamus data digunakan untuk mendefinisikan data yang mengalir dalam sistem dengan lengkap. Kamus data sistem pendukung keputusan proses perbaikan *dies* yang diusulkan antara lain sebagai berikut:

1. Spesifikasi Tabel User

Nama Tabel : user

Fungsi : Untuk menyimpan data user

Tipe : *File data master*

Tabel V.12 Tabel User

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1	<i>Id user</i>	Id_user	<i>Integer</i>	3	<i>Primary Key</i>
2	Nama	Nama	<i>Varchar</i>	25	
3	<i>Username</i>	Username	<i>Varchar</i>	15	
4	<i>Password</i>	Password	<i>Varchar</i>	8	
5	Jabatan	Jabatan	<i>Varchar</i>	25	

(Sumber: Hasil Analisis, 2019)

2. Spesifikasi Tabel *Dies Master*

Nama Tabel : *dies_mst*

Fungsi : Untuk menyimpan data *dies*

Tipe : *File data master*

Tabel V.13 Tabel *Dies Master*

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1	ID <i>Dies</i>	Id_ <i>dies</i>	<i>Varchar</i>	10	<i>Primary Key</i>
2	Nama <i>Dies</i>	Nama_ <i>dies</i>	<i>Varchar</i>	20	
3	Customer	customer	<i>Varchar</i>	20	

(Sumber: Hasil Analisis, 2019)

3. Spesifikasi Tabel *Dies Detail*

Nama Tabel : *dies_dtl*

Fungsi : Untuk menyimpan data *dies*

Tipe : *File data master*

Tabel V.14 Tabel *Dies* Detail

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1	ID <i>Dies</i> Detail	Id_ <i>dies</i> _dtl	<i>Integer</i>	10	<i>Primary Key</i>
2	ID <i>Dies</i>	Id_ <i>dies</i>	<i>Varchar</i>	10	<i>Foreign Key</i>
3	No Proses	No_proses	<i>Varchar</i>	5	
4	Nama Proses	Nama_proses	<i>Varchar</i>	20	

(Sumber: Hasil Analisis, 2019)

4. Spesifikasi Tabel Rekanan

Nama Tabel : rekanan

Fungsi : Untuk menyimpan data rekanan

Tipe : *File data master*

Tabel V.15 Tabel Rekanan

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1	ID Rekanan	Id_rekanan	<i>Varchar</i>	10	<i>Primary Key</i>
2	Nama Rekanan	Nama_rekanan	<i>Varchar</i>	20	
3	Alamat	Alamat	<i>Varchar</i>	100	
4	No Telephone	No_telp	<i>Varchar</i>	15	
5	Email	Email	<i>Varchar</i>	30	

(Sumber: Hasil Analisis, 2019)

5. Spesifikasi Tabel Kerusakan

Nama Tabel : Kerusakan

Fungsi : Untuk menyimpan data kerusakan

Tipe : *File data master*

Tabel V.16 Tabel Kerusakan

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1	ID Kerusakan	Id_kerusakan	<i>Varchar</i>	10	<i>Primary Key</i>
2	Solusi	Solusi	<i>Varchar</i>	200	
3	Kriteria1	<i>Bolt</i>	<i>Varchar</i>	5	
4	Kriteria2	<i>Stopper</i>	<i>Varchar</i>	5	

Tabel V.16 Tabel Kerusakan (lanjutan)

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
5	Kriteria3	<i>Coil_spring</i>	<i>Varchar</i>	5	
6	Kriteria4	<i>Guide_post</i>	<i>Varchar</i>	5	
7	Kriteria5	<i>Die_plat</i>	<i>Varchar</i>	5	
8	Bobot	Bobot	<i>Varchar</i>	5	
9	Keterangan	Ket	<i>Varchar</i>	30	

(Sumber: Hasil Analisis, 2019)

6. Spesifikasi Tabel Perbaikan

Nama Tabel : perbaikan

Fungsi : Untuk menyimpan data perbaikan

Tipe : *File data transaksi*

Tabel V.17 Tabel Perbaikan

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1	ID Perbaikan	Id_perbaikan	<i>Varchar</i>	10	<i>Primary Key</i>
2	Tanggal Kerusakan	Tgl_kerusakan	<i>Date</i>		
3	ID Dies	Id_dies	<i>Varchar</i>	10	<i>Foreign Key</i>
4	ID Dies Detail	Id_dies_Dtl	<i>Varchar</i>	10	<i>Foreign Key</i>
5	ID Kerusakan	Id_kerusakan	<i>Varchar</i>	10	<i>Foreign Key</i>
6	Tanggal perbaikan	Tgl_perbaikan	<i>Date</i>		
7	Perbaikan oleh	Perbaikan_oleh	<i>Varchar</i>	20	
8	Status	Status	<i>Varchar</i>	20	

(Sumber: Hasil Analisis, 2019)

7. Spesifikasi Tabel Perbaikan Rekanan

Nama Tabel : perbaikan_rekanan

Fungsi : Untuk menyimpan data perbaikan rekanan

Tipe : *File data transaksi*

Tabel V.18 Tabel Perbaikan Rekanan

No.	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1	ID Perbaikan Rekan	Id_perbaikan_rekanan	<i>Varchar</i>	10	<i>Primary Key</i>
2	ID Perbaikan	Id_perbaikan	<i>Varchar</i>	10	<i>Foreign Key</i>
3	ID Rekanan	Id_rekanan	<i>Varchar</i>	10	<i>Foreign Key</i>

(Sumber: Hasil Analisis, 2019)

5.10 Perancangan Antarmuka Sistem Pendukung Keputusan Proses Perbaikan Dies

Perancangan antarmuka sistem pendukung keputusan proses perbaikan dies yang diusulkan antara lain sebagai berikut:

1. *Form Login*

Form ini digunakan untuk masuk ke dalam program aplikasi. Rancangan *Form Login* dapat dilihat pada Gambar V.27 berikut:

The image shows a login form with a light beige background. At the top, the word 'LOGIN' is centered in a dark font. Below it, there are two input fields: one labeled 'USERNAME' and another labeled 'PASSWORD'. Both fields have a light blue border. At the bottom center, there is a button labeled 'Login' with a light blue border and a slight shadow effect.

Gambar V.27 *Form Login*
(Sumber: Hasil Analisis, 2019)

2. *Form Halaman Utama*

Form ini merupakan tampilan awal pada aplikasi yang berisi menu-menu.

Rancangan *Form* Halaman Utama dapat dilihat pada Gambar V.28 berikut:

Gambar V.28 *Form* Halaman Utama
(Sumber: Hasil Analisis, 2019)

3. *Form Data Dies*

Form ini digunakan untuk mengelola data *dies*. Rancangan *Form* Data *Dies* dapat dilihat pada Gambar V.29 berikut:

Gambar V.29 *Form* Data *Dies*
(Sumber: Hasil Analisis, 2019)

4. *Form Tambah dan Ubah Data Dies*

Form ini digunakan untuk menambah dan mengubah data *dies*. Rancangan *Form Tambah dan Ubah Data Dies* dapat dilihat pada Gambar V.30 berikut:

Gambar V.30 *Form Tambah dan Ubah Data Dies*
(Sumber: Hasil Analisis, 2019)

5. *Form Data Kerusakan*

Form ini digunakan untuk mengelola data kerusakan. Rancangan *Form Data Kerusakan* dapat dilihat pada Gambar V.31 berikut:

Gambar V.31 *Form Data Kerusakan*
(Sumber: Hasil Analisis, 2019)

6. *Form* Tambah dan Ubah Data Kerusakan

Form ini digunakan untuk menambah dan mengubah data kerusakan. Rancangan *Form* Tambah dan Ubah Data Kerusakan dapat dilihat pada Gambar V.32 berikut:

TAMBAH DATA KERUSAKAN

ID KERUSAKAN

NAMA KERUSAKAN

SOLUSI

BOBOT

KETERANGAN

UBAH DATA KERUSAKAN

NAMA KERUSAKAN

SOLUSI

BOBOT

KETERANGAN

Gambar V.32 *Form* Tambah dan Ubah Data Kerusakan
(Sumber: Hasil Analisis, 2019)

7. *Form* Data Rekanan

Form ini digunakan untuk mengelola data rekanan. Rancangan *Form* Data Rekanan dapat dilihat pada Gambar V.33 berikut:

NUSA INDAH

ADMIN ▾

- HOME
- DATA MASTER ▾
- DIES
- KERUSAKAN
- REKANAN
- PERBAIKAN ▾
- PERMINTAAN PERBAIKAN ▾
- LAPORAN ▾

DATA REKANAN

TAMBAH

ID REKANAN	NAMA REKANAN	ALAMAT	NO TELEPHONE	EMAIL	ACTION

Gambar V.33 *Form* Data Rekanan
(Sumber: Hasil Analisis, 2019)

8. *Form Tambah dan Ubah Data Rekanan*

Form ini digunakan untuk menambah dan mengubah data rekanan. Rancangan *Form* Tambah dan Ubah Data Rekanan dapat dilihat pada Gambar V.34 berikut:

TAMBAH DATA REKANAN

ID REKANAN

NAMA REKANAN

ALAMAT

NO TELEPHONE

EMAIL

UBAH DATA REKANAN

NAMA REKANAN

ALAMAT

NO TELEPHONE

EMAIL

Gambar V.34 *Form* Tambah dan Ubah Data Rekanan
(Sumber: Hasil Analisis, 2019)

9. *Form Validasi Perbaikan*

Form ini digunakan untuk melihat informasi *validasi* perbaikan. Rancangan *Form Validasi* Perbaikan dapat dilihat pada Gambar V.35 berikut:

NUSA INDAH
ADMIN ▾

HOME

DATA MASTER ▾

PERBAIKAN

VALIDASI PERBAIKAN

SELESAI DIPERBAIK

PERMINTAAN PERBAIKAN ▾

LAPORAN ▾

VALIDASI PERBAIKAN

ID PERBAIKAN	TANGGAL KERUSAKAN	NAMA DIES	NAMA KERUSAKAN	TANGGAL PERBAIKAN	TANGGAL PERBAIKAN	ACTION

Gambar V.35 *Form Validasi* Perbaikan
(Sumber: Hasil Analisis, 2019)

10. *Form* Selesai Diperbaiki

Form ini digunakan untuk melihat data *dies* selesai diperbaiki. Rancangan *Form* Selesai Diperbaiki dapat dilihat pada Gambar V.36 berikut:

ID PERBAIKAN	TANGGAL KERUSAKAN	NAMA DIES	NAMA KERUSAKAN	TANGGAL PERBAIKAN	TANGGAL PERBAIKAN	ACTION

Gambar V.36 *Form* Selesai Diperbaiki
(Sumber: Hasil Analisis, 2019)

11. *Form* Permintaan Perbaikan

Form ini digunakan untuk melihat permintaan perbaikan. Rancangan *Form* Permintaan Perbaikan dapat dilihat pada Gambar V.37 berikut:

ID PERBAIKAN	TANGGAL KERUSAKAN	NAMA DIES	NAMA KERUSAKAN	ACTION

Gambar V.37 *Form* Permintaan Perbaikan
(Sumber: Hasil Analisis, 2019)

12. *Form* Tambah Permintaan Perbaikan

Form ini digunakan untuk menambah data perbaikan *dies*. Rancangan *Form* Tambah Permintaan Perbaikan dapat dilihat pada Gambar V.38 berikut:

TAMBAH DATA PERBAIKAN

ID PERBAIKAN

TANGGAL KERUSAKAN

NAMA DIES

NAMA KERUSAKAN

SIMPAN KEMBALI

Gambar V.38 *Form* Tambah Permintaan Perbaikan
(Sumber: Hasil Analisis, 2019)

13. *Form* Permintaan Disetujui

Form ini digunakan untuk melihat permintaan perbaikan yang disetujui. Rancangan *Form* Permintaan Disetujui dapat dilihat pada Gambar V.39 berikut:

NUSA INDAH ADMIN

HOME

DATA MASTER

PERBAIKAN

PERMINTAAN PERBAIKAN

PERMINTAAN PERBAIKAN

PERBAIKAN DISETUJUI

LAPORAN

PERMINTAAN DISETUJUI

ID PERBAIKAN	TANGGAL KERUSAKAN	NAMA DIES	NAMA KERUSAKAN	STATUS	ACTION

Gambar V.39 *Form* Permintaan Disetujui
(Sumber: Hasil Analisis, 2019)

14. *Form* Laporan Perbaikan

Form ini digunakan untuk memilih laporan perbaikan yang ingin dicetak.

Rancangan *Form* Laporan Perbaikan dapat dilihat pada Gambar V.40 berikut:

Gambar V.40 *Form* Laporan Perbaikan
(Sumber: Hasil Analisis, 2019)

15. *Form* Cetak Laporan Perbaikan

Form ini digunakan untuk mencetak laporan perbaikan. Rancangan *Form*

Cetak Laporan Perbaikan dapat dilihat pada Gambar V.41 berikut:

Gambar V.41 *Form* Cetak Laporan Perbaikan
(Sumber: Hasil Analisis, 2019)

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan hasil sistem yang telah dirancang mengenai sistem pendukung keputusan proses perbaikan *dies* pada PT Nusa Indah Jaya Utama, dapat disimpulkan beberapa hal, yaitu:

1. Dengan adanya sistem pendukung keputusan proses perbaikan *dies* dapat membantu divisi *maintenance dies* dalam pengambilan keputusan proses perbaikan *dies*, sehingga para *staff* dari divisi *maintenance dies* dapat langsung mengetahui proses perbaikan apa saja yang harus dilakukan.
2. Sistem yang telah dibangun ini dapat membantu divisi *maintenance dies* dalam mencatat hasil kegiatan perbaikan *dies* sekaligus dapat membantu dalam membuat laporan hasil kegiatan perbaikan *dies*.

6.2 Saran

Saran yang dapat diberikan untuk pengembangan sistem pendukung keputusan proses perbaikan *dies* ini yaitu:

1. Mengimplementasikan hasil dari penelitian berupa sistem pendukung keputusan proses perbaikan *dies* di divisi *Maintenance Dies* pada PT Nusa Indah Jaya Utama.
2. Untuk pengembangan sistem informasi, pada penelitian selanjutnya diharapkan dapat terintegrasi dengan divisi lainnya seperti divisi *maintenance machine*.

DAFTAR PUSTAKA

- Ansori, N dan Mustajib, M Imron. 2013. Sistem Perawatan Terpadu. Edisi 1, Yogyakarta : Graha Ilmu.
- Azmi, Hilman. 2011. Perancangan dan Analisis Stamping Dies untuk Pembuatan Produk Bracket Bumper Dengan Proses Press Multi Forging. Fakultas Teknik Industri. Universitas Gunadarma. Depok.
- Dennis, Alan. 2012. *System Analysis and Design With UML Version 2.0 : An Object Oriented Approach*. Edisi 2. New York : John Wiley & Sons Ltd.
- Fahrozi, Wirhan. 2016. Penerapan Metode Analytical Hierarchy Process (AHP) Dalam Menentukan Ras Ayam Serama, Citec Journal Volume 3, No.3
- Faisal, dan Permana, Silvester Dian Handy. 2015. Sistem Penunjang Keputusan Pemilihan Sekolah Menengah Kejuruan Teknik Komputer Jaringan Yang Terfavorit Dengan Menggunakan Multi-Criteria Decision Making. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK) Volume 2, No.1*.
- Jogiyanto, M. 2005. Analisa dan Desain Sistem Informasi. Yogyakarta : Andi Yogyakarta.
- Kadir, Abdul. 2003. Pengenalan Sistem Informasi. Yogyakarta : Andi Yogyakarta.
- Kadir, Abdul. 2008. Tuntunan Praktis Belajar Database Menggunakan MySQL. Yogyakarta : C.V Andi Offset.
- Kusumadewi, S. et al. 2006. Fuzzy Multi-Attribute Decision Making (FUZZY MADM). Yogyakarta : Graha Ilmu.
- Nugroho, Bunafit. 2007. Membuat Sitem Informasi Pernjualan Berbasis WEB dengan PHP dan MySQL. Yogyakarta : Gava Media.
- Rusdiana, H.A dan Irfan, Moch. 2014. Sistem Informasi Manajemen. Bandung : Pustaka Setia.
- Simatupang, Togar M. 1994. Pemodelan Sistem. Bandung : Nindita Kristen.
- Situmorang, Harold. 2015. Sistem Pendukung Keputusan Pemilihan Calon Peserta Olimpiade Sains Tingkat Kabupaten Langkat Pada Madrasah Aliyah

- Negeri (MAN) 2 Tanjung Pura Dengan menggunakan Metode Simple Additive Weighting (SAW). Jurnal TIMES , Vol. IV, No. 2, hlm. 24-30.
- Subri, Ahmad. 2011. *Rancang Bangun Sistem Penunjang Keputusan Penentuan Mustahik Dengan Pendekatan Analytical Hierarchy Process (AHP) (Studi Kasus: Kantor Pusat Pos Keadilan Peduli Umat (PKPU))*. Fakultas Sains dan Teknologi. Universitas Islam Negeri Syarif Hidayatullah. Jakarta.
- Sutabri, Tata. 2012. *Analisis Sistem Informasi*. Yogyakarta : Andi Yogyakarta.
- Yakub. 2012. *Pengantar Sistem Informasi*. Yogyakarta : Graha Ilmu.
- Rika Yunitarini, 2013, *Sistem Pendukung Keputusan Pemilihan Penyiar Radio Terbaik*, Universitas Trunojoyo Madura, Jurnal Ilmiah Mikrotek Volume 1, No.1.

LAMPIRAN

LAMPIRAN A

URAIAN KEGIATAN PENELITIAN



POLITEKNIK STMI JAKARTA

d.h. SEKOLAH TINGGI MANAJEMEN INDUSTRI

Jl. Letjen Suprpto No. 26 Cempaka Putih, Jakarta 10510

Telp: (021) 42886064 Fax: (021) 42888206

www.stmi.ac.id



DAFTAR HADIR KEGIATAN PRAKTIK KERJA LAPANGAN

1	NAMA MAHASISWA	ABDUR RONI
2	NIM	1315062
3	PROGRAM STUDI	SISTEM INFORMASI INDUSTRI OTOMOTIF

1	NAMA PERUSAHAAN	PT NUSA INDAH JAYA UTAMA
2	ALAMAT/ TELEPON/ EMAIL	JL. LASKAR NO. 49, PEKAYON JAYA, BEKASI – SELATAN
3	DEPARTMEN/ UNIT KERJA	MAINTENANCE DIES
4	NAMA PEMBIMBING/ HP	H. SAIPUDIN. DIPL. ING

No	Department	PIC	JADWAL							
			AGUSTUS				SEPTEMBER			
			1	2	3	4	1	2	3	4
1	PPC Department	M. Syarief								
2	QC Department	Edi Supriadi								
3	Prod Department	Hermawan								
4	MTC Department	Acing Maulana								
5	Adm Department	H. Saipudin								

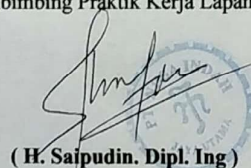
Tanggal	Uraian Kegiatan	Waktu Pulang	Waktu Datang
06/08/2018	Melakukan Pengamatan Proses Perencanaan Produksi Pada Divisi PPC	7 : 30	16 : 30
07/08/2018	Melakukan Pengamatan Proses Perencanaan Produksi Pada Divisi PPC	7 : 30	16 : 30
08/08/2018	Melakukan Pengamatan Proses Perencanaan Pengiriman Produk Pada Divisi PPC	7 : 30	16 : 30
09/08/2018	Melakukan Pengamatan Proses Perencanaan Pengiriman Produk Pada Divisi PPC	7 : 30	16 : 30
10/08/2018	Melakukan Pengamatan Proses <i>Quality Control Before Production</i> Pada Divisi QC	7 : 30	16 : 30
13/08/2018	Melakukan Pengamatan Proses <i>Quality Control On Production</i> Pada Divisi QC	7 : 30	16 : 30



SAI GLOBAL CERTIFICATION SERVICES Pty.Ltd Registration ISO 9001:2008 No.Reg QEC 264727

14/08/2018	Melakukan Pengamatan Proses <i>Quality Control On Production</i> Pada Divisi QC	7 : 30	16 : 30
15/08/2018	Melakukan Pengamatan Proses <i>Quality Control After Production</i> Pada Divisi QC	7 : 30	16 : 30
16/08/2018	Melakukan Pengamatan Proses <i>Quality Control After Production</i> Pada Divisi QC	7 : 30	16 : 30
20/08/2018	Melakukan Pengamatan Proses Produksi <i>Stamping Parts</i> Pada Divisi Produksi	7 : 30	16 : 30
21/08/2018	Melakukan Pengamatan Proses Produksi <i>Stamping Parts</i> Pada Divisi Produksi	7 : 30	16 : 30
23/08/2018	Melakukan Pengamatan Proses Produksi <i>Coating</i> Pada Divisi Produksi	7 : 30	16 : 30
24/08/2018	Melakukan Pengamatan Proses Produksi <i>Coating</i> Pada Divisi Produksi	7 : 30	16 : 30
27/08/2018	Melakukan Pengamatan Proses <i>Maintenance</i> Mesin Pada Divisi Maintenance	7 : 30	16 : 30
28/08/2018	Melakukan Pengamatan Proses <i>Maintenance</i> Mesin Pada Divisi Maintenance	7 : 30	16 : 30
29/08/2018	Melakukan Pengamatan Proses <i>Maintenance Dies</i> Pada Divisi Maintenance	7 : 30	16 : 30
30/08/2018	Melakukan Pengamatan Proses <i>Maintenance Dies</i> Pada Divisi Maintenance	7 : 30	16 : 30
31/08/2018	Melakukan Pengamatan Proses <i>Recruitment</i> Pada Divisi HRD	7 : 30	16 : 30
03/09/2018	Melakukan Pengamatan Proses <i>Marketing</i> Pada Divisi Marketing	7 : 30	16 : 30
04/09/2018	Melakukan Pengamatan Proses <i>Accounting</i> Pada Divisi Accounting	7 : 30	16 : 30
05/09/2018	Melakukan Pengamatan Proses <i>Accounting</i> Pada Divisi Accounting	7 : 30	16 : 30
06/09/2018	Melakukan Pengamatan Proses <i>Accounting</i> Pada Divisi Accounting	7 : 30	16 : 30

Bekasi,
Pembimbing Praktik Kerja Lapangan



(H. Saipudin. Dipl. Ing)



SAI GLOBAL CERTIFICATION SERVICES Pty.Ltd Registration ISO 9001:2008 No.Reg QEC 264727

LAMPIRAN B

JADWAL OBSERVASI

Tanggal	Jam	Pengamat	Yang diamati (Jabatan)	Lokasi Pengamatan	Kegiatan Yang Diamati	Hasil Pengamatan
28-08-2018	09:00	Abdur Roni	Bapak Sunarno (Leader Divisi Maintenance Dies)	Ruang Divisi Maintenance Dies	Proses perbaikan <i>dies</i>	Cara Perbaikan Dies dan Alat – Alat Yang Digunakan
29-08-2018	09:00	Abdur Roni	Bapak Sunarno (Leader Divisi Maintenance Dies)	Ruang Divisi Maintenance Dies	Proses perbaikan <i>dies</i> lebih rinci	Bagian – Bagian <i>Dies</i> dan Dokumen Terkait Perbaikan <i>Dies</i>

Gambar IV.2 Struktur Organisasi PT Nusa Indah Jaya Utama
(Sumber: PT Nusa Indah Jaya Utama, 2018)

LAMPIRAN C

JADWAL WAWANCARA

Tanggal	Jam	Pewawancara	Narasumber (Jabatan)	Lokasi Wawancara	Topik Wawancara
28-08-2018	09:30	Abdur Roni	Bapak Sunarno (Leader Divisi Maintenance Dies)	Ruang Divisi Maintenance Dies	Tugas Divisi <i>Maintenance Dies</i>
28-08-2018	13:30	Abdur Roni	Bapak Sunarno (Leader Divisi Maintenance Dies)	Ruang Divisi Maintenance Dies	Proses Perbaikan <i>Dies</i>
29-08-2018	10:00	Abdur Roni	Bapak Sunarno (Leader Divisi Maintenance Dies)	Ruang Divisi Maintenance Dies	Bagian – Bagian Dies
29-08-2018	14:00	Abdur Roni	Bapak Sunarno (Leader Divisi Maintenance Dies)	Ruang Divisi Maintenance Dies	Alur Proses Bisnis Divisi <i>Maintenance Dies</i>
20-08-2019	13:00	Abdur Roni	Bapak Sunarno (Leader Divisi Maintenance Dies)	Ruang Divisi Maintenance Dies	Konfirmasi dan Pendalaman Proses Bisnis Divisi <i>Maintenance Dies</i>

LAMPIRAN D

TRANSKRIP WAWANCARA

Nama Narasumber : Sunarno (Leader Divisi Maintenance Dies)
Tanggal : 29 Agustus 2018
Jam : 14.00 – 15.00 WIB
Tempat Wawancara : Ruang Divisi *Maintenance Dies*
Topik Wawancara : Alur Proses Bisnis Divisi *Maintenance Dies*

Pewawancara : Pak boleh bertanya-tanya seputar maintenance dies?
Narasumber : Boleh kok mas, mau nanya apa ya?
Pewawancara : Untuk proses perbaikan dies itu bagaimana ya pak alurnya?
Narasumber : Untuk perbaikan dies itu biasanya leader produksi memberikan info bahwa ada dies yang rusak mas.
Pewawancara : Biasanya pemberian infonya itu melalui lisan doang atau bagaimana ya pak?
Narasumber : Untuk pemberian infonya biasanya staff produksi memberi tau kepada leader produksi yang nantinya divisi produksi memberikan info kerusakan melalui form.
Pewawancara : Formnya seperti apa ya pak?
Narasumber : Ada kok mas, sebentar ya saya carikan dulu contoh formnya.
Pewawancara : Untuk perbaikan dies sehari-harinya dilakukan pada perusahaan pak?
Narasumber : Tergantung mas kadang ada beberapa kerusakan yang perusahaan tidak bisa lakukan perbaikan, ya mas bisa lihatlah peralatan perbaikan diperusahaan cuma ada

seadanya.

- Pewawancara : Lalu apabila terjadi kerusakan yang perusahaan tidak bisa lakukan perbaikan bagaimana pak ?
- Narasumber : Biasanya perusahaan akan memberitahukan kepada customer mas apabila terjadi kerusakan yang tidak bisa dikerjakan oleh perusahaan.
- Pewawancara : Kalau sudah diberitahukan kepada customer selanjutnya bagaimana pak?
- Narasumber : Biasanya customer akan kasih keputusan mas, dies yang rusak mau dibenarkan oleh mereka atau dibenarkan oleh rekanan perusahaan.
- Pewawancara : Untuk kerusakan dies biasanya dapat dideteksi berdasarkan apa saja pak?
- Narasumber : Untuk kerusakan dies biasanya dapat dideteksi dari part-part yang ada pada dies mas.
- Pewawancara : Part-partnya apa saja ya pak kalau boleh tau.
- Narasumber : Partnya sama seperti yang mas lihat saat saya melakukan perbaikan kok. Ada Die, Punch, Spring, Guide Post sama Dies Plate mas.
- Pewawancara : Oh seperti yang tadi pagi ya pak.
- Narasumber : Iya mas. Ada lagi yang ingin ditanyakan mas?
- Pewawancara : Untuk sekarang cukup itu dulu kok pak. Terimakasih banyak pak maaf nih kalau mengganggu bapak.
- Narasumber : Iya sama-sama mas, gapapa kok mas tidak mengganggu. Kalau ada yang ingin ditanyakan lagi datang lagi aja kesaya mas.
- Pewawancara : Baik pak nanti kalau ada pertanyaan saya datang lagi ke bapak atau gak ke bang bana pak.
- Narasumber : Oh iya mas ke bana juga gapapa kok mas.
- Pewawancara : Oke pak terimakasih banyak nih pak.

LAMPIRAN E

LISTING PROGRAM

Controllers Dies_Mst

```
<?php
class Dies_Mst extends CI_Controller
{
    public function __construct()
    {
        parent::__construct();
        $this->load->model('M_dies_mst');
        $this->load->library('form_validation');
    }
    public function index()
    {
        $data['record'] = $this->M_dies_mst->tampilkan_data();
        $this->template->load('template', 'dies_mst/lihat_data' , $data);
    }
    public function tambah()
    {
        $data_detail['record'] = $this->M_dies_mst->tampilkan_data_detail();
        $this->template->load('template', 'dies_mst/form_input', $data_detail);
    }
    function detail()
    {
        $id = $this->uri->segment(3);
        $data['record'] = $this->M_dies_mst->tampilkan_detail_by_id_dies($id)-
>result();
        $data['dies_mst'] = $this->M_dies_mst->tampilkan_data_id_dies($id)-
>row_array();
        $this->template->load('template', 'dies_mst/detail_data', $data);
    }
    public function simpan()
    {
        $data['record'] = $this->M_dies_mst->tampilkan_data_detail();
        $this->template->load('template', 'dies_mst/form_input', $data);
    }
    function selesai() {
        $id_dies = $this->M_dies_mst->kode();
        $nama_dies = $this->input->post('nama_dies');
        $customer = $this->input->post('customer');
```

```

        $data = array('id_dies' => $id_dies,'nama_dies' => $nama_dies
,'customer' => $customer);
        $this->M_dies_mst->simpan($data);

        $data_dies = $this->M_dies_mst->last_id_dies_dtl()->row_array();
        $id_dies_dtl = $data_dies['id_dies_dtl'];
        $this->M_dies_mst->selesai($id_dies);
        //alert
        $this->session->set_flashdata('message', '<div class="alert alert-success"
role="alert">
                Data Berhasil Ditambahkan <button type="button" class="close"
data-dismiss="alert" aria-label="Close">
                <span aria-hidden="true">&times;</span>
                </button>
                </div>');
        redirect('Dies_Mst');
    }
    function simpan_detail(){
        $no_proses = $this->input->post('no_proses');
        $nama_proses = $this->input->post('nama_proses');
        $data = array('id_dies' => "0", 'no_proses' => $no_proses , 'nama_proses'
=> $nama_proses);
        $this->M_dies_mst->simpan_detail($data);
        $this->session->set_flashdata('message', '<div class="alert alert-success"
role="alert">
                Data Berhasil Ditambahkan <button type="button"
class="close" data-dismiss="alert" aria-label="Close">
                <span aria-hidden="true">&times;</span>
                </button>
                </div>');
        redirect('Dies_Mst/simpan');
    }
    function hapus_detail()
    {
        $id = $this->uri->segment(3);
        $this->db->delete('dies_dtl', ['id_dies_dtl' => $id]);
        $this->session->set_flashdata('message', '<div class="alert alert-danger"
role="alert">
                Data Berhasil Dihapus <button type="button"
class="close" data-dismiss="alert" aria-label="Close">
                <span aria-hidden="true">&times;</span>
                </button>
                </div>');
    }

```

```

        redirect('Dies_Mst/simpan');
    }
    function hapus(){
        $id = $this->uri->segment(3);
        $this->M_dies_mst->hapus($id);
        //alert
        $this->session->set_flashdata('message', '<div class="alert alert-danger"
role="alert">
                Data Berhasil Dihapus <button type="button"
class="close" data-dismiss="alert" aria-label="Close">
                <span aria-hidden="true">&times;</span>
                </button>
                </div>');
        redirect('Dies_Mst');
    }
    function ubah(){
        if (isset($_POST['submit'])) {
            $id_dies = $this->input->post('id_dies');
            $nama_dies = $this->input->post('nama_dies');
            $customer = $this->input->post('customer');

            $data = array('id_dies' => $id_dies, 'nama_dies' => $nama_dies,
'customer' => $customer);
            $this->M_dies_mst->update($data, $id);
            //alert
            $this->session->set_flashdata('message', '<div class="alert alert-success"
role="alert">
                Data Berhasil Diubah <button type="button"
class="close" data-dismiss="alert" aria-label="Close">
                <span aria-hidden="true">&times;</span>
                </button>
                </div>');
            redirect('Dies_Mst/ubah/'.$id_dies);
        } else {
            $id = $this->uri->segment(3);
            $id_dies = $this->uri->segment(4);
            $data['record'] = $this->M_dies_mst-
>tampilkan_detail_by_id_dies($id)->result();
            $data['dies_mst'] = $this->M_dies_mst-
>tampilkan_data_id_dies($id)->row_array();
            $this->template->load('template', 'dies_mst/form_edit',$data);
        }
    }
    function hapus_detail_ubah(){

```

```

        $id = $this->uri->segment(3);
        $id_dies = $this->uri->segment(4);
        $this->db->delete('dies_dtl', ['id_dies_dtl' => $id]);
        $this->session->set_flashdata('message', '<div class="alert alert-danger"
role="alert">
                Data Berhasil Dihapus <button type="button"
class="close" data-dismiss="alert" aria-label="Close">
                <span aria-hidden="true">&times;</span>
                </button>
                </div>');
        redirect('Dies_Mst/ubah/'.$id_dies);
    }
    function simpan_detail_ubah(){
        $id_dies = $this->input->post('id_dies');
        $no_proses = $this->input->post('no_proses');
        $nama_proses = $this->input->post('nama_proses');
        $data_detail = array('id_dies' => $id_dies, 'no_proses' => $no_proses ,
'nama_proses' => $nama_proses);
        $this->M_dies_mst->simpan_detail($data_detail);
        $this->session->set_flashdata('message', '<div class="alert alert-success"
role="alert">
                Data Berhasil Ditambahkan <button type="button"
class="close" data-dismiss="alert" aria-label="Close">
                <span aria-hidden="true">&times;</span>
                </button>
                </div>');
        redirect('Dies_Mst/ubah/'.$id_dies);
    }
}

```

Models Dies_Mst

```

<?php
class M_dies_mst extends CI_model {
    public function last_id_dies_dtl()
    {
        //untuk menampilkan data
        $query = "SELECT * FROM dies_mst ORDER BY id_dies DESC LIMIT 1";
        return $this->db->query($query);
    }
    function selesai($id_dies)
    {
        $query = "UPDATE `dies_dtl` SET `id_dies`= '$id_dies' WHERE
id_dies = '0'";
        return $this->db->query($query);
    }
}

```

```

    }

    public function kode()
    {
        $q = $this->db->query("select MAX(RIGHT(id_dies,3)) as
code_max from dies_mst");
        $code = "";
        if($q->num_rows()>0){
            foreach($q->result() as $cd){
                $tmp = ((int)$cd->code_max)+1;
                $code = sprintf("%03s", $tmp);
            }
        }else{
            $code = "001";
        }
        return "DSNJ".$code;
    }

    public function tampilkan_data()
    {
        $query="SELECT * FROM dies_mst";
        return $this->db->query($query);
    }

    public function tampilkan_data_detail()
    {
        $query="SELECT * FROM dies_dtl WHERE dies_dtl.id_dies = '0'";
        return $this->db->query($query);
    }

    public function simpan($data)
    {
        $this->db->insert('dies_mst', $data);
    }

    function simpan_detail($data_detail){
        $this->db->insert('dies_dtl', $data_detail);
    }

    public function hapus($id)
    {
        $this->db->where('id_dies', $id);
        $this->db->delete('dies_mst');
        $this->db->where('id_dies', $id);
        $this->db->delete('dies_dtl');
    }

    public function get_data($id)
    {
        $vans = array('id_dies' => $id);

```

```

        return $this->db->get_where('dies_mst', $vans);
    }
    //detail menu
    function tampilkan_detail_by_id_dies($id_dies){
        $query = "SELECT * FROM dies_mst, dies_dtl WHERE
dies_mst.id_dies = dies_dtl.id_dies AND dies_dtl.id_dies = '$id_dies'";
        return $this->db->query($query);
    }
    //ubah menu
    function tampilkan_data_id_dies($id_dies){
        $query = "SELECT * FROM dies_mst WHERE dies_mst.id_dies =
'$id_dies'";
        return $this->db->query($query);
    }
    function update($data, $id)
    {
        $this->db->where('id_dies', $id);
        $this->db->update('dies_mst', $data);
    }
}

```

Views Lihat_Data Dies_Mst

```

<!-- DataTables Example -->
<h2>Data Dies</h2>
<div class="card shadow mb-4">
<div class="card-header py-3">
    <a href="<?php echo site_url(); ?>Dies_Mst/tambah" class="btn btn-primary">Tambah
Data Dies</a>
</div>
<div class="card-body">
    <div class="table-responsive">
        <?php echo $this->session->flashdata('message');?>
        <table class="table table-bordered" id="dataTable" width="100%" cellspacing="0">
        <thead>
            <tr>
                <th>Id Dies</th>
                <th>Nama Dies</th>
                <th>Customer</th>
                <th>Action</th>
            </tr>
        </thead>
        <tbody>
            <?php
                $no = 1;

```



```

foreach ($record->result() as $b) {
    echo "<tr>
    <td>$b->id_dies</td>
    <td>$b->nama_dies</td>
    <td>$b->customer</td>
    <td>
        ".anchor('Dies_Mst/ubah/'.$b->id_dies,'Ubah',array('class' => 'btn btn-success'))."
        ".anchor('Dies_Mst/detail/'.$b->id_dies,'Detail',array('class' => 'btn btn-primary'))."
        ".anchor('Dies_Mst/hapus/'.$b->id_dies,'Hapus',array('class' => 'btn btn-danger'))."
    </td>
    </tr>";
    $no++;
}
?>
</tbody>
</table>
</div>

<!-- Modal Input -->
<div class="modal fade" id="tambah" tabindex="-1" role="dialog" aria-
labelledby="exampleModalLabel" aria-hidden="true">
    <div class="modal-dialog" role="document">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title" id="exampleModalLabel">Tambah Data Dies</h5>
                <button type="button" class="close" data-dismiss="modal" aria-label="Close">
                    <span aria-hidden="true">&times;</span>
                </button>
            </div>
            <div class="modal-body">
                <div class="card-body">
                    <? =
                        form_open('Dies_Mst/simpan');
                    ?>
                    <div class="box-body">
                        <div class="form-group">
                            <label>Id Dies</label>
                            <input type="text" name="id_dies" class="form-control" id=""
placeholder="ID Dies" required="true">
                        </div>
                        <div class="form-group">
                            <label>Nama Dies</label>
                            <input type="text" name="nama_dies" class="form-control" id=""
placeholder="Nama Dies" required="true">

```

```

    </div>
    <div class="form-group">
      <label>Customer</label>
      <input type="text" name="customer" class="form-control" id=""
placeholder="Customer" required="true">
    </div>

    </div><!-- /.box-body -->
    <button type="button" class="btn btn-secondary" data-
dismiss="modal">Kembali</button>
    <button type="submit" class="btn btn-primary">Simpan</button>
  </div>
  <div class="modal-footer">
  </div>
</div>
</div>
</div>

```

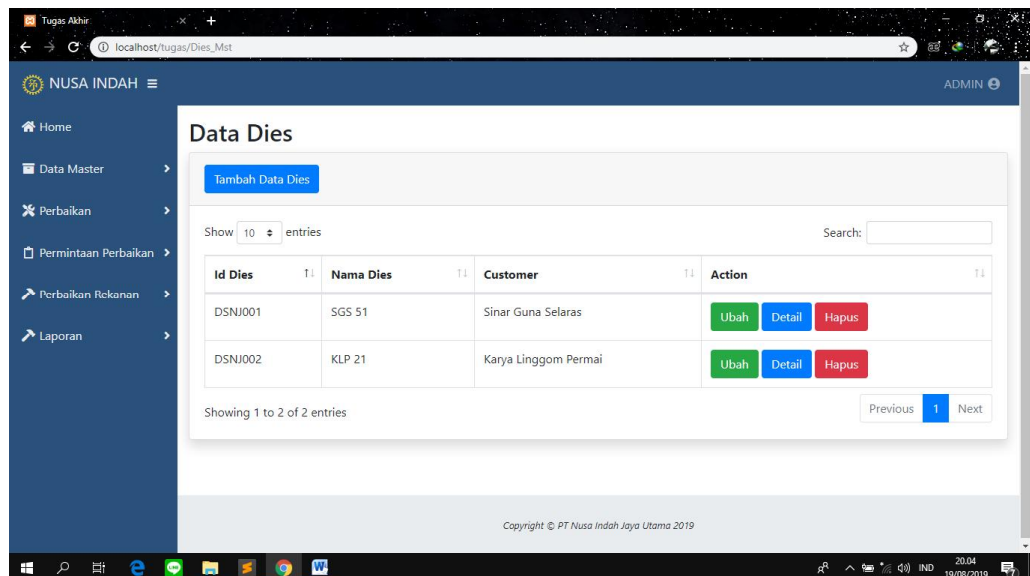
LAMPIRAN F

TAMPILAN PROGRAM

1. Home



2. Data Master



3. Perbaikan

Data Perbaikan

Show 10 entries Search:

Id Perbaikan	Tanggal Kerusakan	Nama Dies	Nama Kerusakan	Solusi	Keterangan	Status	Action
PRBKN005	2019-08-09	SGS 51 - Piercing	Hasil Dies Burry	Grinding Dies	Tidak Perlu Penganan Khusus	Belum Diperbaiki	Konfirmasi Detail

Showing 1 to 1 of 1 entries Previous 1 Next

Copyright © PT Nusa Indah Jaya Utama 2019

4. Permintaan Perbaikan

Data Permintaan Perbaikan

[Tambah Data Perbaikan](#)

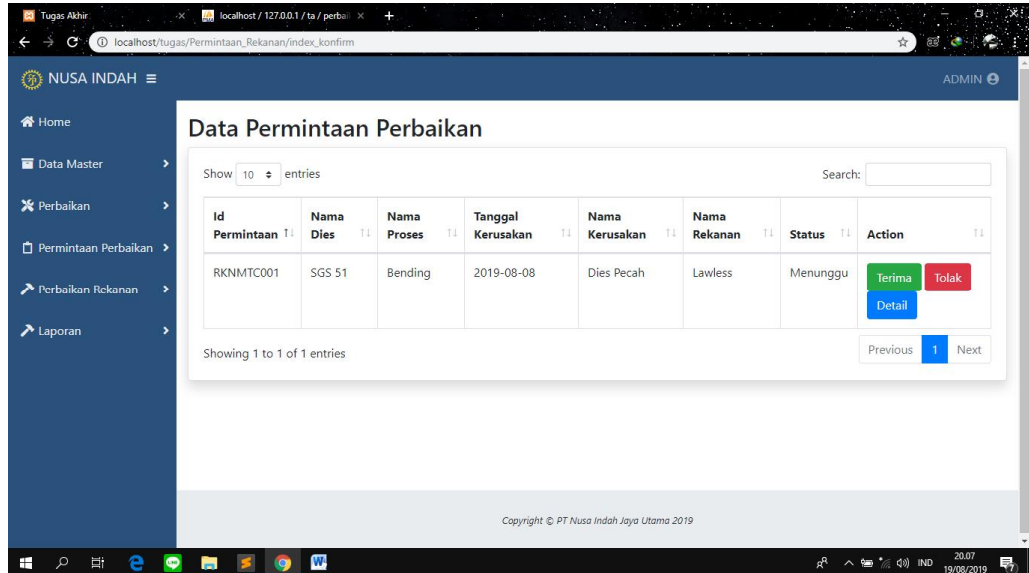
Show 10 entries Search:

Id Perbaikan	Tanggal Kerusakan	Nama Dies	Nama Proses	Nama Kerusakan	Action
PRBKN005	2019-08-09	SGS 51	Piercing	Hasil Dies Burry	Hapus

Showing 1 to 1 of 1 entries Previous 1 Next

Copyright © PT Nusa Indah Jaya Utama 2019

5. Perbaikan Rekanan



Data Permintaan Perbaikan

Show 10 entries

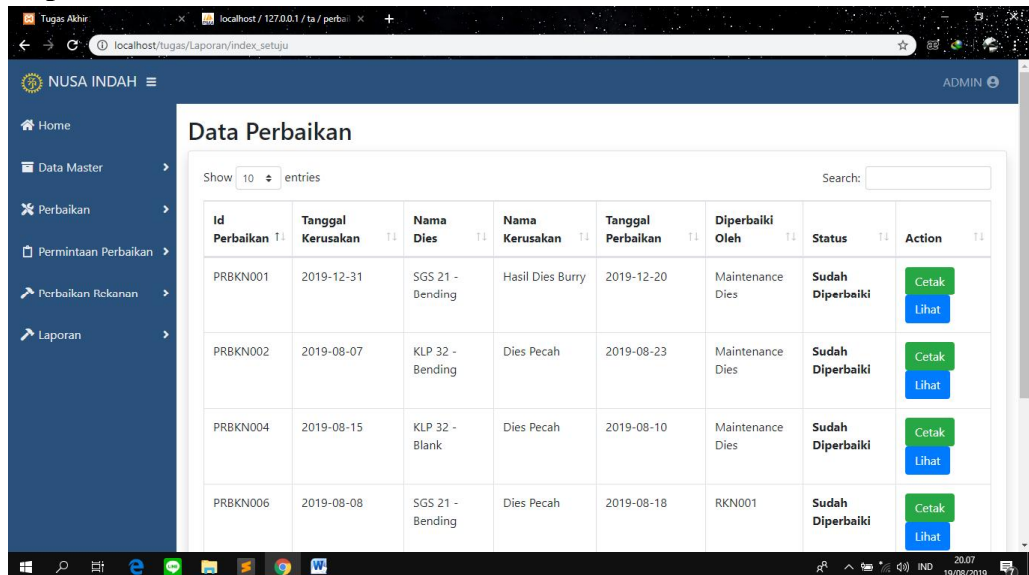
Id Permintaan	Nama Dies	Nama Proses	Tanggal Kerusakan	Nama Kerusakan	Nama Rekanan	Status	Action
RKNMTC001	SGS 51	Bending	2019-08-08	Dies Pecah	Lawless	Menunggu	Terima Tolak Detail

Showing 1 to 1 of 1 entries

Previous 1 Next

Copyright © PT Nusa Indah Jaya Utama 2019

6. Laporan



Data Perbaikan

Show 10 entries

Id Perbaikan	Tanggal Kerusakan	Nama Dies	Nama Kerusakan	Tanggal Perbaikan	Diperbaiki Oleh	Status	Action
PRBKN001	2019-12-31	SGS 21 - Bending	Hasil Dies Burry	2019-12-20	Maintenance Dies	Sudah Diperbaiki	Cetak Lihat
PRBKN002	2019-08-07	KLP 32 - Bending	Dies Pecah	2019-08-23	Maintenance Dies	Sudah Diperbaiki	Cetak Lihat
PRBKN004	2019-08-15	KLP 32 - Blank	Dies Pecah	2019-08-10	Maintenance Dies	Sudah Diperbaiki	Cetak Lihat
PRBKN006	2019-08-08	SGS 21 - Bending	Dies Pecah	2019-08-18	RKN001	Sudah Diperbaiki	Cetak Lihat

LAMPIRAN G

BLACK-BOX TESTING

Penguji : Bagas Eko Saputro

NIM : 1315051

Aktivitas Pengujian	Realisasi yang diharapkan	Hasil Pengujian	Kesimpulan
Memasukkan <i>username</i> dan <i>password</i> dengan benar lalu mengklik tombol <i>Login</i>	Sistem menerima akses <i>login</i> dan menampilkan menu utama	Masuk ke menu utama	[x] Diterima [] Ditolak
Klik <i>Logout</i>	Sistem akan keluar dan kembali ke halaman <i>login</i>	Kembali ke halaman <i>login</i>	[x] Diterima [] Ditolak
Memilih menu <i>data master</i> dan sub-menu kelola <i>dies</i>	Sistem akan menampilkan menu kelola <i>dies</i>	Muncul menu untuk mengelola data <i>dies</i>	[x] Diterima [] Ditolak
Klik tambah data <i>dies</i>	Sistem akan memunculkan menu data <i>dies</i>	Muncul menu input data <i>dies</i>	[x] Diterima [] Ditolak
Klik selesai dengan menginput data yang dibutuhkan pada menu input data <i>dies</i>	Sistem akan menyimpan data dan menampilkannya pada menu data <i>dies</i>	Data yang diinputkan muncul pada menu data <i>dies</i>	[x] Diterima [] Ditolak
Klik tombol hapus pada menu data <i>dies</i>	Sistem akan menghapus kolom yang dipilih untuk	Kolom akan terhapus sesuai pilihan yang	[x] Diterima [] Ditolak

Aktivitas Pengujian	Realisasi yang diharapkan	Hasil Pengujian	Kesimpulan
	dihapus	dihapus	
Klik tombol ubah pada menu data <i>dies</i>	Sistem akan menampilkan menu ubah data <i>dies</i>	Muncul menu ubah data <i>dies</i>	[x] Diterima [] Ditolak
Klik selesai dengan mengubah data yang akan diubah pada menu ubah data <i>dies</i> .	Sistem akan mengubah data dan menampilkannya pada menu data <i>dies</i>	Data yang diubah akan muncul pada menu data <i>dies</i>	[x] Diterima [] Ditolak
Klik detail pada menu data <i>dies</i>	Sistem akan menampilkan detail data <i>dies</i> yang dipilih berdasarkan baris	Muncul detail data dari data <i>dies</i> yang dipilih berdasarkan baris	[x] Diterima [] Ditolak
Memilih menu permintaan perbaikan dan sub-menu buat permintaan	Sistem akan menampilkan permintaan perbaikan <i>dies</i>	Muncul menu untuk mengelola data permintaan perbaikan <i>dies</i>	[x] Diterima [] Ditolak
Klik tambah data perbaikan	Sistem akan memunculkan menu input data permintaan perbaikan	Muncul menu input data permintaan perbaikan	[x] Diterima [] Ditolak
Klik simpan dengan menginput data yang dibutuhkan pada menu input data permintaan perbaikan <i>dies</i>	Sistem akan menyimpan data dan menampilkannya pada menu data permintaan perbaikan <i>dies</i>	Data yang diinputkan muncul pada menu data permintaan perbaikan <i>dies</i>	[x] Diterima [] Ditolak
Klik tombol hapus pada	Sistem akan menghapus kolom	Kolom akan terhapus sesuai	[x] Diterima [] Ditolak

Aktivitas Pengujian	Realisasi yang diharapkan	Hasil Pengujian	Kesimpulan
menu data permintaan perbaikan <i>dies</i>	yang dipilih untuk dihapus	pilihan yang dihapus	
Memilih menu perbaikan dan sub-menu perbaikan rekanan	Sistem akan menampilkan perbaikan <i>dies</i>	Muncul menu untuk mengelola data perbaikan <i>dies</i>	[x] Diterima [] Ditolak
Klik tambah data perbaikan <i>dies</i>	Sistem akan memunculkan menu input data perbaikan	Muncul menu input data perbaikan	[x] Diterima [] Ditolak
Klik simpan dengan menginput data yang dibutuhkan pada menu input data perbaikan <i>dies</i>	Sistem akan menyimpan data dan menampilkannya pada menu data perbaikan <i>dies</i>	Data yang diinputkan muncul pada menu data perbaikan <i>dies</i>	[x] Diterima [] Ditolak
Klik tombol hapus pada menu data perbaikan <i>dies</i>	Sistem akan menghapus kolom yang dipilih untuk dihapus	Kolom akan terhapus sesuai pilihan yang dihapus	[x] Diterima [] Ditolak
Klik detail pada menu data perbaikan <i>dies</i>	Sistem akan menampilkan detail data perbaikan <i>dies</i> yang dipilih berdasarkan baris	Muncul detail data dari data perbaikan <i>dies</i> yang dipilih berdasarkan baris	[x] Diterima [] Ditolak
Memilih menu perbaikan dan sub-menu konfirmasi perbaikan	Sistem akan menampilkan konfirmasi perbaikan <i>dies</i>	Muncul menu untuk mengelola data konfirmasi perbaikan <i>dies</i>	[x] Diterima [] Ditolak
Klik detail pada menu data konfirmasi perbaikan	Sistem akan menampilkan detail data konfirmasi perbaikan <i>dies</i>	Muncul detail data dari data konfirmasi perbaikan <i>dies</i>	[x] Diterima [] Ditolak

Aktivitas Pengujian	Realisasi yang diharapkan	Hasil Pengujian	Kesimpulan
perbaikan <i>dies</i>	yang dipilih berdasarkan baris	yang dipilih berdasarkan baris	
Klik konfirmasi pada menu data konfirmasi perbaikan <i>dies</i>	Sistem akan mengubah status data perbaikan <i>dies</i> yang dipilih berdasarkan baris	Mengubah status menjadi Sudah Diperbaiki pada data perbaikan <i>dies</i>	[x] Diterima [] Ditolak
Memilih menu laporan dan sub-menu selesai diperbaiki	Sistem akan menampilkan data selesai diperbaiki	Muncul menu untuk mengelola data laporan selesai diperbaiki	[x] Diterima [] Ditolak
Klik cetak pada menu data laporan selesai diperbaiki	Sistem akan mencetak data laporan selesai diperbaiki yang dipilih berdasarkan baris	Mencetak laporan sesuai dengan baris yang dipilih.	[x] Diterima [] Ditolak